On the Design of Adaptive Fuzzy Regression Selection Filter with the Application of Image Restoration

Pao-Ta Yu and Jia-Hong Wu

Department of Computer Science and Information Engineering
National Chung Cheng University
Chiayi, Taiwan 62107, R.O.C.
E-mail: csipty@ccunix.ccu.edu.tw, Fax: 886-5-272-0859

Abstract

In this paper we present a detection-estimation filter based on the technique of adaptive fuzzy regression. The method of fuzzy regression is useful in data estimation and is more flexible than the conventional regression methods. However, the time to calculate an optimal solution in fuzzy regression analysis prohibits us to apply the fuzzy regression method directly to the field of image restoration. Thus, we present a new algorithm to approximate the optimal solution of the fuzzy linear regression method. This new method, called the adaptive fuzzy linear regression method, will meet the need of speed in image restoration. The *Adaptive Fuzzy Regression Selection* (*AFRS*) filter combines the technique of the adaptive fuzzy linear regression with the *RCRS* filter. The central pixel of the window in the *AFRS* filter is replaced with the estimated output of the *RCRS* filter only if the central pixel is corrupted. A model is formed using the adaptive fuzzy linear regression method to decide whether the central pixel is corrupted. The computer simulations prove that the *AFRS* filter is useful in image restoration.

1 Introduction

In the field of image restoration, many nonlinear filters have been introduced to provide better noise cancellation. There is an important class of nonlinear filters named the selection filter which select their result from one of the input pixels in a window. The median filter and the rank conditioned rank selection (*RCRS*) filter are both selection filters [1]. These filters have a drawback of replacing the central pixel with one of the pixels in the window, regardless of whether the central pixel is corrupted or not. Recently, some filters based on a detection-estimation strategy were introduced in the field of filter design [2, 3]. These filters prove to be more efficient than the conventional filters and the results are better. These filters select the output value of a pixel between the original value and the estimated value depending on the detection of an impulse noise.

In this paper, we introduce a new detection-estimation style filter called the *Adaptive Fuzzy Regression Selection* (*AFRS*) filter. The *AFRS* filter combines the technique of the adaptive fuzzy linear regression and the *RCRS* filter. The method of the adaptive fuzzy linear regression approximates the optimal solution of the fuzzy linear regression. In the *AFRS* filter a neuron based on the adaptive fuzzy linear regression method will be used for noise-detection and the *RCRS* filter will be used for estimation of true value of the pixels that are corrupted.

In the next section, the adaptive fuzzy linear regression algorithm will be introduced. In Section 3, the structure of the *AFRS* filter will be defined. In Section 4, the filtering capability of the *AFRS* filter will be compared with the filtering capability of the *RCRS* filter, the *SD-ROM* filter and the fuzzy filter. Finally, in Section 5 the conclusion of this new filter will be made.

2 The Adaptive Fuzzy Linear Regression Algorithm

Conventional regression methods like the Least-Square (LS) method are useful in data estimation. They treat the difference of the estimated values and the observed values as observation errors, and the difference is considered a random variable [4]. The fuzzy regression methods view these difference as ambiguity inherently present in the system. The optimal solution of the fuzzy linear regression is often calculated by the method of linear programming, but the computation time to find the solution is not polynomial. Thus it is not possible to use fuzzy linear regression in the field of image processing. In this paper, we propose the adaptive fuzzy linear regression algorithm to approximate the optimal solution of the fuzzy linear regression. The adaptive fuzzy linear regression algorithm can solve the problem in a bounded time, thus we found it suitable to apply the algorithm to the field of image processing.

First we define the structure of the neuron used in this algorithm. The input-output relation of the adaptive fuzzy linear regression is defined as

$$Y = \sum_{k=1}^{n} W_k x_k \qquad (1)$$

where the coefficients W_k and the estimated output Y are fuzzy numbers and the inputs x_k are crisp numbers. The desired output is defined as T, which is also a fuzzy number. We define the fuzzy numbers used in this paper to be

$$\mu_A(x) = \exp\left(-\frac{(x - r_A)^2}{2\sigma_A^2}\right) \qquad (2)$$

where r_A is the center of the fuzzy number and σ_A is the deviation.

The objective of this algorithm is to minimize the deviation of the estimated fuzzy number under the constraint that the degree of fitting of the estimated fuzzy number should be larger than a certain confidence level H. The degree of fitting of an estimated fuzzy number h can be defined as

$$h = \sup_h \mu_{Th} \subseteq \mu_{Yh} \qquad (3)$$

where μ_{Th}, μ_{Yh} are h-level sets [4]. The estimated output Y of the fuzzy model can be derived as

$$\mu_Y(x) = \exp\left(-\frac{(x - (\sum_{k=1}^{n} r_{W_k} x_k))^2}{2(\sum_{k=1}^{n} \sigma_{W_k}|x_k|)^2}\right)$$

which in turn yields

$$h = \exp\left(-\frac{(r_T - (\sum_{k=1}^{n} r_{W_k} x_k))^2}{2((\sum_{k=1}^{n} \sigma_{W_k}|x_k|) - \sigma_T)^2}\right). \qquad (4)$$

The goal of the algorithm can be reached by using two sets of weight update rule. The goal of the first set of weight update rule is that the degree of fitting of every sample should be larger than the confidence level H. By differentiating h and using the fact that omitting h in the weight update rule won't affect the direction that the coefficients are adjusted, we can obtain the following two gradients:

$$\nabla_{r_{W_k}} h = \frac{(r_T - (\sum_{k=1}^{n} r_{W_k} x_k))}{(\sigma_T + (\sum_{k=1}^{n} |\sigma_{W_k} x_k|))^2} x_k \qquad (5)$$

$$\nabla_{\sigma_{W_k}} h = \frac{(r_T - (\sum_{k=1}^{n} r_{W_k} x_k))^2}{(\sigma_T + (\sum_{k=1}^{n} |\sigma_{W_k} x_k|))^3} |x_k| \qquad (6)$$

for $k = 1, 2, ..., n$.

From (5) and (6) we can obtain the first set of weight update rule of the learning algorithm.

$$r_{W_k}(n+1) = r_{W_k}(n) + \eta_1 \nabla_{r_{W_k}} h(n) \qquad (7)$$

$$\sigma_{W_k}(n+1) = \sigma_{W_k}(n) + \eta_2 \nabla_{\sigma_{W_k}} h(n) \qquad (8)$$

for $k = 1, 2, ..., n$.

The objective of the second weight update rule of the learning algorithm is to minimize the deviation of the estimated fuzzy number if the degree of fitting is larger than the confidence level H. The second set of weight update rule of the learning algorithm can be easily obtained as:

$$\sigma_{W_k}(n+1) = (1 - \eta_3 x_k)\sigma_{W_k}(n) \tag{9}$$

for $k = 1, 2, ..., n$.

3 The Structure of the *AFRS* Filter

The structure of the *AFRS* filter contains two parts, the adaptive fuzzy regression neural selector and the *RCRS* filter. The output of the filter is defined as

$$y(n) = \gamma R(n) + (1 - \gamma)x(n)$$

where $R(n)$ is the output of the *RCRS* filter, and γ is a parameter that is decided by the adaptive fuzzy regression neural selector. The structure of *AFRS* filter is shown in the following figure.

First we define the structure of the adaptive fuzzy regression neural selector, which uses a neuron based on the adaptive fuzzy linear regression algorithm described in Section 2. The difference of

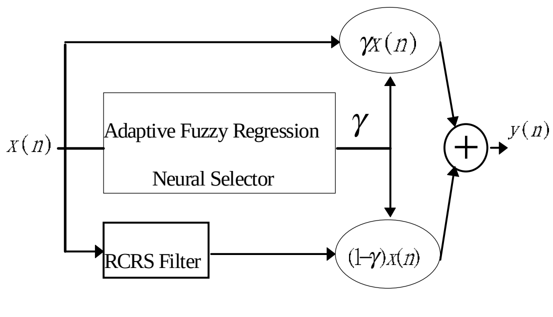

Figure 1: Structure of AFRS filter.

the pixel in the original image and the corresponding pixel in the noisy image is fuzzyfied to obtain the desire output T. This is done by adding a deviation to form a fuzzy number. The fuzzyfied desire output has membership function of

$$\mu_T(x) = \exp\left(-\frac{(x - (L_{noise} - L_{origin})^2}{2\sigma^2}\right)$$

where L is the grey level of a pixel in the corrupted or the original image and σ is an arbitrary small positive number.

The neural selector is designed in a recursive style similar to the recursive median filter. For a 3×3 window centered at $(i,\ j)$, the input vector is defined as

$$\begin{aligned}\underline{x} \;=\; & (y_{i-1,\,j-1},\ y_{i-1,\,j},\ y_{i-1,\,j+1}, y_{i,\,j-1}, \\ & x_{i,\,j},\ x_{i,\,j+1},\ x_{i+1,\,j-1},\ x_{i+1,\,j},\ x_{i+1,\,j+1})\end{aligned}$$

where x is the pixel fetched from the noisy image. As for y, when the *AFRS* filter is in the learning phase, it is fetched from the original image. When the *AFRS* filter is in the filtering phase, it is fetched from the previously filtered pixels.

The selection function that calculates the coefficient γ is defined as

$$\gamma = sel(r_Y)$$

where r_Y is the center of the estimated output fuzzy number Y and sel is the selection function. The selection function can be shown in Figure 2. There are four thresholds which separate the region of uncorrupted pixels from the regions of pixels corrupted by positive impulse noise and negative impulse noise. The thresholds are defined in a fuzzy style which allows the coefficient γ to be of value from 0 to 1. The thresholds are defined as a, b, c and d with $a > b > c > d$.

The *RCRS* filter used in this model is a recursive style *RCRS* filter. The definition of the input

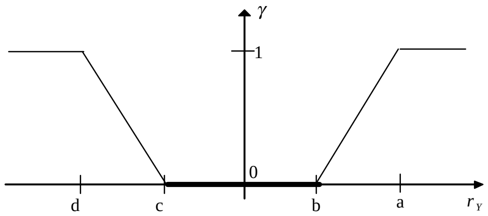

Figure 2: Selection function used by AFRS filter.

	AFRS	*RCRS*	Fuzzy	*SD-ROM*
10%	23.802	47.4	36.483	31.53
20%	47.659	89.747	65.289	69.17
30%	66.890	149.0	94.399	138.07
40%	95.599	248.541	139.431	236.85

Table 1: MSE Value of Different Noise Probability

vector of the *RCRS* filter are the same as the input vector of the neural selector. Only the input vectors that the neural selector considered to be corrupted have their corresponding training pairs fed in the *RCRS* filter, and only the input vectors that is considered corrupted are filtered.

4 Experimental Results

In this section we compare the simulated results of the *AFRS* filter with other filters. Among them are the *RCRS* filter, *SD-ROM* filter and the fuzzy filter proposed by Russo and Ramponi, which is a kind of filter that is based on fuzzy rules. These filters have excellent filtering capability. The images used in the experiments are the 256×256 Lenna image in various noise. The sliding window used by these filters are of size 3×3. The *MSE* criterion is used in the experiments.

The neural selector in the *AFRS* filter converges after 5 to 20 epochs of learning. The filtering results of the *AFRS*, *RCRS*, *SD-ROM* and fuzzy filter are shown in Table 1. From the table we can see that the filtering capability of the *AFRS* filter is better than the *RCRS,SD-ROM* or the fuzzy filter.

Figure 3: The original lenna image.

Figure 4: Lenna image corrupted with 30% of noise.

Figure 5: Recovered Lenna image using AFRS.

Figure 6: Recovered Lenna image using FF.

5 Conclusions

In this paper, a new kind of detection-estimation style filter that uses the adaptive fuzzy linear regression algorithm and the *RCRS* filter has been proposed. By combining the adaptive fuzzy linear regression algorithm and the recursive *RCRS* filter, the filtering capability of the *AFRS* filter proves to be good. The simulation results have shown that the *AFRS* filter provides better filtering results than the conventional *RCRS* or the fuzzy filter in various cases, including highly corrupted images.

Figure 7: Recovered Lenna image using RCRS.

References

[1] R. C. Hardie and K. E. Barner, "Rank conditioned rank selection filters for signal restoration," *IEEE Trans. on Image Processing*, vol. 3, no. 2, pp.192-206, March 1994.

[2] E. Abreu and S. K. Mitra, "A signal-dependent rank ordered mean (*SD-ROM*) filter: a new approach for removal of impulses from highly corrupted images," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, ICASSP-95,* Detroit, MI, May 1995.

[3] Fabrizio Russo and Giovanni Ramponi, " Fuzzy Filter for Images Corrupted by Impulse Noise," *IEEE Signal Processing Letters*, vol. 3, pp. 168-170, June 1996.

[4] H. Tanaka, S. Uejima and K. Asai, "Linear Regression Analysis with Fuzzy Model," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-12, no.6, 1982.

Figure 8: Recovered Lenna image using SD-ROM.