VLSI Implementation of Image Sensor for Spatial Filtering

K. Siva¹ and SP. Senthil Kumar²
¹PG Student, M.E (VLSI Design), ²Associate Professor
Department of ECE, Shanmuganathan Engineering College, India

ABSTRACT

A low-complexity adaptive scaling algorithm is proposed for the implementation of 2-D image scaling applications. A less complexity, less memory requirement, and high performance algorithm is proposed for Very Large Scale Integration implementation of an image scaling processor. The anticipated image scaling algorithm consists of a clamp filter, spatial filter and a bilinear interpolation. The spatial and clamp filters are added as pre-filters for reducing the aliasing artifacts resulted by the bilinear interpolation. The sharpening spatial filter is added as a pre-filter to reduce the blurring effect produced by the bilinear interpolation. Furthermore, an adaptive technology is used to enhance the effect of the edge detector by adaptively selecting the input pixels of the bilinear interpolation. In addition, an algebraic manipulation and a hardware sharing techniques are used to simplify bilinear interpolation, which efficiently reduces the computing resources and silicon area in very large scale integration (VLSI). Combined filter is replaced by a dynamic estimation unit to minimize the hardware cost. Compared with Previous methodologies, this work shows better performance with respect to cost and less complexity.

Keywords: Clamp Filter; Bilinear; Spatial Filter; Adaptive Filter

1. INTRODUCTION

In the recent years, image enhancement has become a component of many important image processing and computer vision applications. Image enhancement involves taking an image and improving it visually, typically by taking advantage of the response to visual stimuli. Sequences of enhancement techniques are widely used to facilitate the development of a solution for computer image problems.

Many of these techniques require the use of low illumination or high magnification where problems associated with noise persist. For this reason, noise removal continues to be an important image processing task.

Image noise represents unwanted or undesired information that can occur during the image capture, transmission, processing or acquisition, and may be dependent or independent of the image content. In typical images, the noise can be modeled with either a Gaussian, uniform or salt-and-pepper distribution. A filter is something that attenuates or enhances particular frequencies easiest to visualize in the frequency domain. Image filters have a wide variety of uses such as noise removal and edge enhancement. It is creating an embossed appearance and making an image appear crisper and sharper.

Spatial domain represents an important enhancement technique that can effectively be used to remove various types of noise in digital images. These spatial filters typically operate on small neighborhood 3 x 3 to 11 x 11 and some of them can be implemented as convolution masks. Mean filters are the most commonly spatial filters used as a simple method for reducing noise in an image, particularly Gaussian noise. Each pixel value in an image with the mean ‘average’ value of its neighbors, including itself. The
extracted average values are the result of the convolution process, which is commonly based on specified fixed convolution mask (kernel). Differently sized kernels containing different patterns of number achieve different results under convolution. By increasing the size of the mean filter to 5 x 5, the obtained image will be characterized with less noise and less high frequency detail.

2. SPATIAL DOMAIN

Spatial domain represents an important enhancement technique that can effectively be used to remove various types of noise in digital images. These Spatial filters typically operate on small neighborhood 3 x 3 to 11 x 11 and some of them can be implemented as convolution masks. Mean filters are the most commonly spatial filters used as a simple method for reducing noise in an image, particularly Gaussian noise. The idea of mean filtering is simply to replace each pixel value in an image with the mean ‘average’ value of its neighbors, including itself. The extracted average values are the result of the convolution process, which is commonly based on specified fixed convolution mask (kernel).

Differently sized kernels containing different patterns of number achieve different results under convolution. By increasing the size of the mean filter to 5 x 5, the obtained image will be characterized with less noise and less high frequency detail. In this, two new image enhancement filters have been developed; to remove and enhance the appearance of an image according to the distance measure between adjacent pixels. These filters are Far Distance Filter (FDF) and Near Distance Filter (NDF). Compared to the well-known mean filter, the proposed filters can achieve better results in visual and quantitative measures.

A. Spatial Filtering

Spatial filtering refers to some neighborhood operations working with the values of the image pixels in the neighborhood and the corresponding values of a sub image that has the same dimensions as the neighborhood. This sub image is called, a filter, mask, kernel, template or a window. The value in a filter is referred to as coefficients.

The filtering can be performed in

1. Spatial domain.
2. Frequency domain

B. Spatial-Temporal Invariant Motion

Let us consider first the case where the motion is spatio-temporal invariant, namely,

\[ I(x, y, t) = I(x - V_x t, y - V_y t, 0) = I_0(x - V_x t, y - V_y t) \]

And

\[ \hat{I}(\omega_x, \omega_y, \omega_t) = \hat{I}_0(\omega_x, \omega_y) \delta(\omega_x V_x + \omega_y V_y + \omega_t). \]
Namely, the sampled data energy is restricted to the main plane and its shifted Versions. Clearly, as long as the main plane contains no elements of the reciprocal Lattice, we are guaranteed not to have Aliasing.

That is,

\[ \tilde{V}_x = \frac{\Delta t}{\Delta x} V_x; \quad \tilde{V}_y = \frac{\Delta t}{\Delta y} V_y \]

Which are the velocities (or, rather, the optical flow) expressed in the units (pixels/frame).

Then above equations can be rewritten as

\[ \ell \tilde{V}_x + m \tilde{V}_y + n \neq 0 \quad \forall (\ell, m, n) \neq (0, 0, 0). \]

Or, equivalently,

\[ \ell \tilde{V}_x + m \tilde{V}_y \neq \left[ \ell \tilde{V}_x + m \tilde{V}_y \right] \quad \forall (\ell, m) \neq (0, 0), \]

where \( |a| = \text{round}(a) \) (the closest integer to \( a \in \mathbb{R} \)).

Observing (10) above equation, we note that if either \( v_x \), or \( v_y \) or \( v_x/v_y \), are rational numbers (10) does not hold. This means that, for \( I_0(x,y) \), which is not band limited, aliasing is unavoidable.

When \( I_0(x,y) \) is band limited, say that

\[ \hat{I}_0(\omega_x, \omega_y) = 0 \quad \text{for all} \quad (\omega_x, \omega_y) \text{such that} \quad |\omega_x| > \frac{W_x}{2} \quad \text{or} \quad |\omega_y| > \frac{W_y}{2} \]

Above equations can be written as

\[ \ell \tilde{V}_x + m \tilde{V}_y \neq \left[ \ell \tilde{V}_x + m \tilde{V}_y \right] \quad \forall (\ell, m) \neq (0, 0) \text{ such that } |\ell| \leq \frac{W_x}{\Delta x} \frac{2\pi}{2\pi} \quad \text{and} \quad |m| \leq \frac{W_y}{\Delta y} \frac{2\pi}{2\pi}. \]

Another constructive way of stating above equation is through defining

\[ \beta = \min_{(\ell, m) \neq (0, 0)} \left\{ \ell \tilde{V}_x + m \tilde{V}_y - \left[ \ell \tilde{V}_x + m \tilde{V}_y \right] \right\}. \]

Then, (11) is equivalent to the requirement \( \beta > 0 \). (Note that we have postulated that there is aliasing in a single frame, namely, we have both \( 2\pi/\Delta x < W_x \) and \( 2\pi/\Delta y < W_y \)—otherwise there is no need of SR reconstruction).

The significance of \( b \) will be further discussed later. It is however clear, that it provides a measure of how far one is from aliasing in the ideal case discussed here. Hence, the larger \( b \) is the more robust the assumption of ideal case. Namely, when, as a result of finite image size, the energy of image Fourier Transform (FT) is not restricted to a plane anymore, the size of \( b \) is an indication of what combinations of
data size, sampling intervals and velocities will avoid aliasing. Clearly, from (11) there are velocities which are preferable as far as aliasing and SR reconstruction is concerned.

Furthermore, we observe that the pair \((\ell, m)\) leading to \(\beta\) identify the closest relevant shifted plane. Hence, \((2\pi/\Delta r)\beta\) is the distance between the closest relevant shifted plane and the main plane. This leads to the following possible reconstruction procedure.

Assuming that \(\beta > 0\) the original image can be reconstructed by passing the data through the filter defined by its frequency response

\[
\tilde{h}(\omega_x, \omega_y, \omega_z) = \text{rect}_r(\omega_x) \text{rect}_r(\omega_y) \text{rect}_r(\omega_z V_x + \omega_y V_y + \omega_z),
\]

where

\[
\text{rect}_r(x) = \begin{cases} 
1 & \text{for } |x| \leq \frac{\ell}{2}, \\
0 & \text{otherwise},
\end{cases}
\]

And, using our observation earlier, we allow half the minimal distance around the main plane.

Namely, we choose,

\[
W_r = \frac{2\pi}{\Delta r} \beta.
\]

Then, the result will be

\[
\tilde{I}_{\text{eq}}(\omega_x, \omega_y, \omega_z) = \tilde{h}(\omega_x, \omega_y, \omega_z) \tilde{I}_{\text{d}}(\omega_x, \omega_y, \omega_z) = \tilde{I}_0(\omega_x, \omega_y) \delta(\omega_x V_x + \omega_y V_y + \omega_z) = \tilde{I}(\omega_x, \omega_y, \omega_z).
\]

Namely, we have achieved perfect reconstruction. This type of filter is referred to in the literature as a motion compensated filter.

The dependence of \(\beta\) on \(W_r, W_y, V_y\) and \(V_z\) (as presented in Eq. (B.1)) is quite intriguing. One could ask, for given \(\Delta r, \Delta y, \Delta v, W_r\) and \(W_y\), what is the largest \(\beta\) possible. Namely, pose the following problem: Find

\[
\beta^* = \max_{V_x, V_y} \min_{(\omega_x, \omega_y) \neq (0, 0)} \left| \left( V_x + m \tilde{V}_y - \left[ \tilde{V}_x + m \tilde{V}_y \right] \right) \right|.
\]

3. SHARPENING SPATIAL AND CLAMP FILTERS

The sharpening spatial and clamp filters act as pre-filters to reduce blurring and aliasing artifacts produced by the bilinear interpolation. The input pixels of the original images are filtered by the spatial filter to remove associated noise and enhance the edges. Unwanted discontinuous edges and boundaries are filtered by clamp filter. To conserve computing resource and memory buffer, these two filters are simplified and combined into a combined filter.

![Fig. 3 Shows The Block Diagram of the Image Scaling Algorithm.](image-url)
4. SHARPENING SPATIAL FILTER

The sharpening spatial filter is a kind of high-pass filter, is used to enhance the edges as well as the details of objects, and is extremely effective at removing associated noise. The kernel of the sharpening spatial filter is designed to increase the brightness of the center pixel relative to neighboring pixels. It usually contains a single positive value at its center and completely surrounded by negative values. The following array is an example of a $3 \times 3$ kernel for a sharpening spatial filter

$$Kernel_s = \begin{bmatrix}
-1 & -1 & -1 \\
-1 & S & -1 \\
-1 & -1 & -1
\end{bmatrix}$$

Where $S$ is a sharp parameter that can be set according to the characteristics of the images. The degree of sharpening can be adjusted by changing the sharp parameter. In this paper, the sharpening spatial filter is used to improve the appearance of images by increasing the delineation between bright and dark regions, which reduces blurring effects caused by bilinear interpolation. As with the clamp filter, results show that the filtered mage is clearer than the original one.

A. Less-Complexity Sharpening Spatial and Clamp Filters

The sharpening spatial filter, a kind of high-pass filter, is used to reduce blurring artifacts and defined by a kernel to increase the intensity of a center pixel relative to its neighboring pixels. The clamp filter a kind of low pass filter is a 2-D Gaussian spatial domain filter and composed of a convolution kernel array.

It usually contains a single positive value at the center and is completely surrounded by ones. The clamp filter is used to reduce aliasing artifacts and smooth the unwanted discontinuous edges of the boundary regions. The sharpening spatial and clamp filters can be represented by convolution kernels. A larger size of convolution kernel will produce higher quality of images. However, a larger size of convolution filter will also demand more memory and hardware cost.

For example, a $6 \times 6$ convolution filter demands at least a five-line-buffer memory and 36 arithmetic units, which is much more than the two-line-buffer memory and nine arithmetic units of a $3 \times 3$ convolution filter. In our previous work, each of the sharpening spatial and clamp filters was realized by a 2-D $3 \times 3$ convolution kernel as shown in Fig. It demands at least a four-line-buffer memory for two $3 \times 3$ convolution filters.

In the proposed scaling algorithm, both the T-model and inverse T-model filters are used to improve the quality of the images simultaneously. The T-model or inverse T-model filter is simplified from the $3 \times 3$ convolution filter of the previous work, which not only efficiently reduces the complexity of the convolution filter but also greatly decreases the memory requirement from two to one line buffer for each convolution filter.

The T-model and the inverse T-model provide the low-complexity and low memory-requirement convolution kernels for the sharpening spatial and clamp filters to integrate the VLSI chip of the proposed low-cost image scaling processor.
B. Combined Filter

In proposed scaling algorithm, the input image is filtered by a sharpening spatial filter and then filtered by a clamp spatial filter again. Although the sharpening spatial and clamp filters are simplified by T-models and inversed T-models, it still needs two line buffers to store input data or intermediate values for each T-model or inversed T-model filter. Thus, to be able to reduce more computing resource and memory requirement, sharpening spatial and clamp filters, which are formed by the T-model or inversed T-model, should be combined together into a combined filter as

\[
P'(m,n) = \left[ \begin{array}{c} \frac{1}{(S-3)} \end{array} \right] \left[ \begin{array}{c} S \ C \\ S \ -1 \ (C+3) \end{array} \right] \left[ \begin{array}{c} P(m,n) \\ -1 \end{array} \right]
\]

Where \(S\) and \(C\) are the sharp and clamp parameters and \(P(m,n)\) is the filtered result of the target pixel \(P(m,n)\) by the combined filter. A T-model sharpening spatial filter and a T-model clamp filter have been replaced by a combined T-model filter as shown in . To reduce the one-line-buffer memory, the only parameter in the third line, parameter \(-1\) of \(P(m,n-2)\), is removed, and the weight of parameter \(-1\) is added into the parameter \(S-C\) of \(P(m,n-1)\) by \(S-C-1\) as shown in (2). The combined inversed T-model filter can be produced in the same way.

In the new architecture of the combined filter, the two T-model or inversed T-model filters are combined into one combined T-model or inversed T-model filter. By this filter-combination technique, the demand of memory can be efficiently decreased from two to one line buffer, which greatly reduces memory access requirements for software systems or hardware memory costs for VLSI implementation.

C. Spatial Enhancement

Spectral enhancement relies on changing the gray scale representation of pixels to give an image with more contrast for interpretation. It applies the same spectral transformation to all pixels with a given gray scale in an image. However, it does not take full advantage of human recognition capabilities even though it may allow better interpretation of an image by a user, Interpretation of an image includes the use of brightness information, and the identification of features in the image. Several examples will demonstrate the value of spatial characteristics in image interpretation. Spatial enhancement is the mathematical processing of image pixel data to emphasize spatial relationships. This process defines homogeneous regions based on linear edges. Spatial enhancement techniques use the concept of spatial frequency within an image. Spatial frequency is the manner in which gray-scale values change relative to their neighbors within an image. If there is a slowly varying change in gray scale in an image from one side of the image to the other, the image is said to have a low spatial frequency. If pixel values vary radically for adjacent pixels in an image, the image is said to have a high spatial frequency.

Spatial enhancement involves the enhancement of either low or high frequency information within an image. Algorithms that enhance low frequency image information employ a "blurring" filter (commonly called a low pass filter) that emphasizes low frequency parts of an image while de-emphasizing the high frequency components. The enhancement of high frequency information within an image is often called edge enhancement. It emphasizes edges in the image while retaining overall image quality.

D. Objectives of Spatial Enhancement Technique

Objectives or Purposes There are three main purposes that underlie spatial enhancement techniques are improve interpretability of image data, aid in automated feature extraction, remove and/or reduce sensor degradation Methods.

The two major methods commonly used in spatial enhancement are:
1. Convolution
2. Fourier Transform

D.1 Convolution

Convolution involves the passing of a moving window over an image and creating a new image where each pixel in the new image is a function of the original pixel values within the moving window and the coefficients of the moving window as specified by the user. The window, a convolution operator, may be considered as a matrix (or mask) of coefficients that are to be multiplied by image pixel values to derive a new pixel value for a resultant enhanced image. This matrix may be of any size in pixels and does not necessarily have to be square.

Examples: As an example of the convolution methodology, take a 3 by 3 matrix of coefficients and see the effects on an example image subset. A set of coefficients that is used for image smoothing and noise removal is given below:

Table 6D. Convolution Coefficients
If we have a sample image, given directly above where the image normally has a low smoothly varying gray scale, except for the bottom right region, which exhibits a sharp brightness change, we can see the effects of the convolution filter on a pixel-by-pixel basis.

Two important examples of image enhancement are:

(i) increasing the contrast, and
(ii) changing the brightness level of an image so that the image looks better.

Image restoration is one of the prime areas of image processing and it is very much objective. The restoration techniques are based on mathematical and statistical models of image degradation. Denoising and Deblurring tasks come under this category. Its objective is to recover the images from degraded observations. The techniques involved in image restoration are oriented towards modeling the degradations and then applying an inverse procedure to obtain an approximation of the original image. Hence, it may be treated as a deconvolution operation. Depending on applications, there are various types of imaging systems. X-ray, Gamma ray, ultraviolet, and ultrasonic imaging systems are used in biomedical instrumentation. In astronomy, the ultraviolet, infrared and radio imaging systems are used. Sonic imaging is performed for geological exploration. Microwave imaging is employed for radar applications. But, the most commonly known imaging systems are visible light imaging.

Such systems are employed for applications like remote sensing, microscopy, measurements, consumer electronics, entertainment electronics, etc. The images acquired by optical, electro-optical or electronic means are likely to be degraded by the sensing environment. The degradation may be in the form of sensor noise, blur due to camera misfocus, relative object camera motion, random atmospheric turbulence, and so on. The noise in an image may be due to a noisy channel if the image is transmitted through a medium. It may also be due to electronic noise associated with a storage-retrieval system. Noise in an image is a very common problem. An image gets corrupted with noise during acquisition, transmission, storage and retrieval processes.

The various types of noise are discussed in the next chapter. Noise may be classified as substitutive noise, (impulsive noise: e.g., salt & pepper noise, random-valued impulse noise, etc.) additive noise (e.g., additive white Gaussian noise) and multiplicative noise (e.g. speckle Noise). The impulse noise of low and moderate noise densities can be removed easily by simple denoising schemes available in the literature. The simple median filter works very nicely for suppressing impulse noise of low density. However, nowadays, many denoising schemes are proposed which are efficient in suppressing impulse noise of moderate and high noise densities. In many occasions, noise in digital images is found to be additive in nature with uniform power in the whole bandwidth and with Gaussian probability distribution. Such a noise is referred to as Additive White Gaussian Noise (AWGN).

It is difficult to suppress AWGN since it corrupts almost all pixels in an image. The arithmetic mean filter, commonly known as Mean filter, can be employed to suppress AWGN but it introduces a blurring effect. Multiplicative (speckle Noise) is an inherent property of medical ultrasound imaging.

Speckle noise occurs in almost all coherent imaging systems such as laser, acoustics and SAR (Synthetic Aperture Radar) imagery. and because of this noise the image resolution and contrast become reduced,
thereby reducing the diagnostic value of this imaging modality. So, speckle noise reduction is an important prerequisite, whenever ultrasound imaging is used for tissue characterization.

In my work I have introduced this speckle noise to considered image and analysed for various spatial and transform domain filters by considering all the image metrics. Among the many methods that have been proposed to perform this task, there exists a class of approaches that use a multiplicative model of speckled image formation and take advantage of the logarithmical transformation in order to convert multiplicative speckle noise into additive noise. The common assumption made in a dominant number of such studies is that the samples of the additive noise are mutually uncorrelated and obey a Gaussian distribution.

5. BILINEAR INTERPOLATION

Bilinear interpolation is an image-restoring algorithm, which linearly interpolates four nearest-neighbor pixels of an unrestored image to obtain the pixel of a restored image as a forward function. The principle behind the bilinear interpolation algorithm is executing a linear interpolation in one direction, and then repeating the same function in the other direction

\[ P(i, j), P(i+1,j), P(i,j+1), \text{ and } P(i+1,j+1) \] are the four nearest neighbor pixels of the original image with \( i = [0, 1, 2, \ldots M] \) and \( j = [0, 1, 2, \ldots N] \).

Here, \( M \) is the number of pixels having the width of the original image and \( N \) is the number of pixels corresponding to the length of the original image.

![Fig.7.1 Bilinear Interpolation.](image)

The temporary pixels \( P(x',j) \) and \( P(x',j+1) \) are created by linear interpolation in horizontal direction and can be calculated as

\[
P(x_,j) = (1 - xf) \times P(i, j) + xf \times P(i+1,j)
\]
\[
P(x_,j+1) = (1 - xf) \times P(i, j+1) + xf \times P(i+1,j+1)
\]

Where \( xf \) is the scale parameter in horizontal direction. After interpolating in horizontal direction, the values of temporary pixels \( P(x_,j) \) and \( P(x_,j+1) \) are generated.

The resulting output pixel \( P(x',y') \) can be obtained by one more linear interpolation in the other direction. Alternatively, the output can be produced by implementing linear interpolation in the vertical direction and can be calculated.

\[
P(x_,y_) = [(1 - xf) \times P(i, j) + x \times P(i+1,j)] \times
(1 - yf) + [(1 - xf) \times P(i, j+1) + xf \times P(i+1,j+1)] \times yf
\]

\[ (29) \]
Where $v_f$ is the scale parameter in vertical direction. Bilinear interpolation is popular in the implementation of VLSI chips due to its low complexity and simple architecture. However, its high-frequency response behavior is poor as a result of linear changes to the output pixel value according to sampling position. Results show that the edges become blurry and the aliasing effect is visible after being processed using bilinear interpolation.

In the proposed scaling algorithm, the bilinear interpolation method is selected because of its characteristics with low complexity and high quality. The bilinear interpolation is an operation that performs a linear interpolation first in one direction and, then again, in the other direction. The output pixel $P(k,l)$ can be calculated by the operations of the linear interpolation in both $x$- and $y$-directions with the four nearest neighbour pixels. We can easily find that the computing resources of the bilinear interpolation cost eight multiply, four subtract, and three addition operations. It costs a considerable chip area to implement a bilinear interpolator with eight multipliers and seven adders. Thus, an algebraic manipulation skill has been used to reduce the computing resources of the bilinear interpolation.

### 5.1 The Problem Statement

Efficient suppression of noise in an image is a very important issue. Denoising finds extensive applications in many fields of image processing. Image Denoising is an important pre-processing task before further processing of image like segmentation, feature extraction, texture analysis etc. The purpose of Denoising is to remove the noise while retaining the edges and other detailed features as much as possible. Conventional techniques of image denoising using linear and nonlinear techniques have already been studied and analyzed for efficient denoising scheme. Here various Spatial and Transform domain filters are considered to denoise the noisy images, having various noise variances.

### 5.2. Background

Filters play a significant role in the image denoising process. It is a technique for modifying or enhancing an image. The basic concept behind reducing noise in noisy images using linear filters is digital convolution and moving window principle.

Linear filtering is filtering in which the value of an output denoised pixel is a linear combination of the values of the pixels in the input pixel's neighborhood. Let $w(x,y)$ be the input signal subjected to filtering, and $z(x)$ be the filtered output.

If the applied filter satisfies certain conditions such as linearity and shift invariance, then the output filter can be expressed mathematically in simple form as given below

$$z(x) = \int w(t)h(x-t)dt$$

Where $h(t)$ is impulse response or point spread function and it completely characterizes the filter. The above process called as convolution and it can be expressed as

$$Z = W \ast h.$$  

In case of discrete convolution the filter is as given below

$$z(i) = \sum_{-\infty}^{\infty} w(t)h(i-t)$$
This means that the output at point \( z(i) \) is given by a weighted sum of input pixels surrounding \( i \) and here the weights are given by \( h(t) \). To create the output at the next pixel \( i+1 \), the function is \( h(t) \) shifted by one and the weighted sum is computed again.

The overall output is created by a series of shift-multiply-sum operations, and this forms a discrete convolution. For the 2-dimensional case, \( h(t) \) becomes \( h(t, u) \), and above Equation

\[
z(i, j) = \sum_{t=-k}^{i+k} \sum_{u=-j}^{j-k} w(t, u) h(i-t, j-u)
\]

Here the values of \( h(t, u) \) are referred to as the filter weights, the filter kernel, or filter mask. For reasons of symmetry \( h(t) \) is always chosen to be of size \( mn \). Where \( m \) and \( n \) are both usually odd (often \( m=n \)). In physical systems the kernel \( h(t, u) \) must be non-negative, which results in some blurring or averaging of the image. The narrower the \( h(t, u) \), then the filter gives less blurring. In digital image processing, \( h(t, u) \) maybe defined arbitrarily and this \( h(t, u) \) gives rise to many types of filters.

6. ADAPTIVE FILTER

![Figure 6. Adaptive Filter](image)

Generally speaking, the adaptive process involves the use of a cost function, which is a criterion for optimum performance of the filter, to feed an algorithm, which determines how to modify filter transfer function to minimize the cost on the next iteration.

As the power of digital signal processors has increased, adaptive filters have become much more common and are now routinely used in devices such as mobile phones and other communication devices, camcorders and digital cameras, and medical monitoring equipment.

6.1 Inputs and Output of a Generic Rls Adaptive Filter

This section presents a brief description of how adaptive filters work and some of the applications where they can be useful. Adaptive filters self learn. As the signal into the filter continues, the adaptive filter coefficients adjust themselves to achieve the desired result, such as identifying an unknown filter or canceling noise in the input signal. In the figure below, the shaded box represents the adaptive filter, comprising the adaptive filter and the adaptive recursive least squares (RLS) algorithm.

![Figure 6.1 Inputs and Output of a Generic RLS Adaptive Filter](image)

6.2 Noise (Or) Interference Cancellation

In noise cancellation, adaptive filters let you remove noise from a signal in real time. Here, the desired signal, the one to clean up, combines noise and desired information. To remove the noise, feed a signal \( n'(k) \) to the adaptive filter that represents noise that is correlated to the noise to remove from the desired signal.
7. CONCLUSION

In this paper, two new spatial image de-noising filters have been developed. These filters work to remove abnormal pixels (noise) from an image by using some calculations that depend on the pixel distance and inverting the distance between pixels. The extracted results obviously illustrate the efficiency of the proposed filters and give better image quality compared to the mean filter, which is used widely in image enhancement of the cardinality of 5 (5 x 5 kernel). It is possible to improve these filters further by adding more criteria, such as threshold, or by expanding the filter kernel cardinality. This is left for future work.

REFERENCES

[6] “A Low-Cost VLSI Implementation For Efficient Removal Of Impulse Noise” Pei-Yin Chen, Member, IEEE, Chih-Yuan Lien, Student Member, Ieee, And Hsu-Ming Chuang.