Image Denoising using new digital pulse mode neural network

Suresh .R (ME)\(^1\), Kannadhasan .G(ME)\(^2\), Jebin.P.L(ME)\(^3\)
Embedded System technologies, S A Engineering College,
Chennai-77, Tamil nadu, India.

Abstract—In this paper, we propose a new architecture of Pulse Mode Neural Network (PMNN) with very simple activation function. ANNs are usually implemented in software, but many applications require implementation of fast and large neural networks on efficient custom device. Pulse mode is gaining support in the field of hardware Neural Networks thanks to its higher density of integration. Hardware implementation is thus devoted only to the generalization phase using the network parameters set. In this context, the main idea is to apply a new kind of activation function, simply generated by the product of two sigmoidal functions, which are very simple and already implemented in previous work. The pulse mode originates from where a stochastic computing is applied using basic logic gates and random pulse sequences as inputs. The filtered results are verified in terms of the Peak Signal to Noise Ratio (PSNR). Experimental results reveal that the proposed PMNN filter has a greater ability to recover the informative pixel intensities from the infected image with a recovery of 7.5 dB for Gaussian noise and 5.3 dB for Speckle noise. In this work, looking for simple implementation and efficient learning, we propose to implement a new kind of adjustable highly modulator function based on simple existing non linear sigmoidal blocks. Besides, such results demonstrate the performance and efficiency of our Neural filter when compared to other conventional filtering techniques.

Key words : PSNR, PMNN, Sigmoidal, Gaussian noise,filter.

1. INTRODUCTION

Image processing is an important component of modern technologies because human depends so much on the visual information than other creatures. Image is better than any other information form for us to perceive. Among our information about the world, 99% is perceived with our eyes. Image processing has traditionally been an area in the engineering community.

Today, the medical industry, astronomy, physics, chemistry, forensics, remote sensing, manufacturing, and defense are just some of the many fields that rely upon images to store, display, and provide information about the world around us. The challenge to scientists, engineers and business people is to quickly extract valuable information from raw image data. This is the primary purpose of image processing – converting images to information.

2. PROJECT DESCRIPTION

Image denoising is an important image processing task, both as a process itself, and as a component in other processes. Very many ways to denoise an image or a set of data exists. The main properties of a good image denoising model is that it will remove noise while preserving edges. Traditionally, linear models have been used. One common approach is to use a Gaussian filter, or equivalently solving the heat equation with the noisy image as input data. In some cases, the activation values of the units undergo a relaxation process such that the network will evolve to a stable state in which these activations do not change anymore.

One big advantage of linear noise removal models is the speed. The linear models is that they are able to preserve edges in a good manner. Edges, which are recognized as discontinuities in the image,. Nonlinear models on the other hand can handle edges in a much better way than linear models. One popular model for nonlinear image denoising is the modified median filter. This filter is very good at preserving edges, but smoothly varying regions in the input image are transformed into piecewise constant regions in the output image.

Block diagram

The basic architecture consists of three types of neuron layers: input, hidden, and output. In feed-forward networks, the signal flow is from input to output units, strictly in a feed-forward direction. The data processing can extend over multiple layers of units, but no feedback connections are present.

2.1. Input mask
Masking is available for selected environmental functions including statistics, classification, unmixing, matched filtering, continuum removal and spectral feature fitting. Use masking to create image masks.

**Selecting Areas:**
Select from the following options to define mask areas
- To set the defined areas in the mask to 1 (On) or to 0 (Off). The mask is built using a Logical OR or Logical AND operation between all of the items in the list.
- To define the mask using only those areas where the listed data ranges, annotation shapes and select Logical AND.
- To use all the defined areas to make the mask, select Logical OR.

### 2.2. Input Layer

Every neural network has the number of neurons comprising this layer, this parameter is completely and uniquely determined once know the shape of the training data. Specifically, the number of neurons comprising that layer is equal to the number of features (columns) of the data. Some neural network configurations add one additional node for a bias term.

### 2.3 Hidden Layer

The hidden layer configuration using just two rules:

(i) Number of hidden layers equals one
(ii) The number of neurons in that layer is the mean of the neurons in the input and output layers.

A set of hidden units which take input from the input layer. The hidden units collectively form the hidden layer. For simplicity, we assume that each unit in the input layer is connected to each unit of the hidden layer, but this isn't necessarily the case. A weighted sum of the output from the input units forms the input to every hidden unit. Note that the number of hidden units is usually smaller than the number of input units.

### 2.4 Output Layer

The output layer like the Input layer, every neural network has exactly one output layer. Determining its size (number of neurons) is simple. It is completely determined by the chosen model configuration.

Neural network is running in Machine Mode or Regression Mode (the ML convention of using a term that is also used in statistics but assigning a different meaning to it is very confusing). If the neural network is a regressor, then the output layer has a single node.

### 3. NEURAL NETWORK SIGMOIDAL FUNCTION

A sigmoid function is a bounded differentiable real function that is defined for all real
input values and that has a positive derivative everywhere.

The key point is that this architecture is very simple and very generalized. This same flow diagram can be used for many problems, regardless of their particular process. The ability of the neural network to provide useful data manipulation lies in the proper selection of the weights. This is a dramatic departure from conventional information processing where solutions are described in step by step procedures.

As an example, imagine a neural network for recognizing objects in a sonar signal. Suppose that 1000 samples from the signal are stored in a computer. How does the computer determine if these data represent a submarine, whale, undersea mountain, or nothing at all? Conventional DSP would approach this problem with mathematics and algorithms, such as correlation and frequency spectrum analysis. With a neural network, the 1000 samples are simply fed into the input layer, resulting in values popping from the output layer. By selecting the proper weights, the output can be configured to report a wide range of information. For instance, there might be outputs for: submarine (yes/no), whale (yes/no), undersea mountain (yes/no), etc.

A sigmoid activation function uses the sigmoid function to determine its activation. The sigmoid function is defined as follows:

\[ f(x) = \frac{1}{1 + e^{-x}} \]

4. DENOISING THE IMAGE

Noise is the result of errors in the image acquisition process that result in pixel values that do not reflect the true intensities of the real scene. There are several ways that noise can be introduced into an image, depending on how the image is created.

The different filter as a denoiser leads to solving a 2nd order nonlinear PDE. Since smooth regions are transformed into piecewise constant regions when using the filter, it is desirable to create a model for which smoothly varying regions are transformed into smoothly varying regions, and yet the edges are preserved. The median is much less sensitive than the mean to extreme values (called outliers).

This can be done for instance by solving a 4th order PDE instead of the 2nd order PDE from the filter. Results show that the 4th order filter produces much better results in smooth regions, and still preserves edges in a very good way.
The three images above show a small excerpt of the normal vectors of the above shown image. The first image shows the normal of the original image, the middle image shows the normal of the noisy image, and the last image shows the smoothed normal.

4.1. Different Types of Noise

Gaussian noise

Gaussian noise is evenly distributed over the signal. This means that each pixel in the noisy image is the sum of the true pixel value and a random Gaussian distributed noise value. As the name indicates, this type of noise has a Gaussian distribution, which has a bell shaped probability distribution function.

Salt and Pepper Noise

Salt and pepper noise [Um98] is an impulse type of noise, which is also referred to as intensity spikes. This is caused generally due to errors in data transmission. It has only two possible values, a and b. The probability of each is typically less than 0.1. The corrupted pixels are set alternatively to the minimum or to the maximum value, giving the image a “salt and pepper” like appearance. Unaffected pixels remain unchanged. For an 8-bit image, the typical value for pepper noise is 0 and for salt noise 255. The salt and pepper noise is generally caused by malfunctioning of pixel elements in the camera sensors, faulty memory locations, or timing errors in the digitization process.

Speckle Noise

Speckle noise is a multiplicative noise. This type of noise occurs in almost all coherent imaging systems such as laser, acoustics and SAR(Synthetic Aperture Radar) imagery. The source of this noise is attributed to random interference between the
coherent returns. Fully developed speckle noise has the characteristic of multiplicative noise.

**Brownian Noise**

Brownian noise comes under the category of fractal or 1/f noises. The mathematical model for 1/f noise is fractional Brownian motion. Fractal Brownian motion is a non-stationary stochastic process that follows a normal distribution. Brownian noise is a special case of 1/f noise. It is obtained by integrating white noise

5. **CONCLUSION AND FUTURE ENHANCEMENTS**

The whole network was implemented in computer and Microcontroller chip. Such implementation offers many advantages over other solution with respect to both hardware implementation cost and device timing performance. Denoising of an image reduces significantly the amount of data and filters out information that may be regarded as less irrelevant. Denoising is efficient in medical imaging. By using floating point number system for synapse weight value representation, any function can be approximated by the network.

The most important feature of the proposed function is the simplicity of implementation and the easy of programmability, not making use of any complex explicit function to be implemented, which increase the learning capacity and enhance the Network efficiency.

6. **REFERENCES**


