DATA FILTRATION AND SIMULATION BY ARTIFICIAL NEURAL NETWORK

Sweety H Meshram†*, Vinay Keswani*, Nilesh Bodne*

*Department of Electronics & Communication, Vidharbha Institute of Technology, RTMNU Nagpur.
†Corresponding author

ABSTRACT

The automotive industry requires an automated system to sort different sizes and shapes objects, images which are the mainly used component in the industry, to improve the overall productivity. There are things at which humans are still way ahead of the machines in terms of efficiency one of such thing is the recognition especially pattern recognition. There are several methods which are tested for giving the machines the intelligence in efficient way for pattern recognition purpose. The artificial neural network is one of the most optimization techniques used for training the networks for efficient recognition. Computer vision is the science and technology of machines that can see. The machine is made by integration of many parts to extract information from an image in order to solve some task. Principle component analysis is a technique that will be suitably used for the application purpose for sorting, inspection, fault diagnosis in various field.

Keywords: Digital Signature, MATLAB, Recognition, Wavelet Transforms, Principle Component Analysis, Artificial Neural Network.

1. INTRODUCTION

The machine is made by integration of many parts to extract information from an image in order to solve some task. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images. Each of the application areas described above employ a range of computer vision tasks; with more or less well defined measurement or processing problems, which can be solved using a variety of methods. Some examples of typical computer vision tasks are presented below. Recognition is the classical problem in computer vision, image processing, and machine vision. It is related to the determination of whether or not the image data contains some specific object, feature, or activity.
The human perception of identifying an object is the natural logical thinking process by which humans recognise an object. But, machines are far behind the human recognition system of an object, so researchers are up-to increasing this efficiency of the machines.

Artificial neural network is the evolutionary technique which mimics the human brain of retaining the data when it is first identified or trained. The main focus of this paper is on the various feature extraction techniques which can be used along with the artificial neural network. The optimization algorithm has less iteration than implementation with Artificial Neural Network process for the same task and other improved algorithms while the convergence rate is faster and the precision is higher. Curve figures in terms of perimeter radius are used as feature extraction for recognizing objects. This method is efficient and more suitable for real time recognition systems compared with previous research because we can get better iteration time, speed of belt conveyor and accuracy.

2. NEED OF ARTIFICIAL NEURAL NETWORK

2.1 Concept of Artificial Neural Network

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. The key element of this paradigm is the novel structure of the information processing system. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems. ANNs, like people, learn by example. An ANN is configured for a specific application, such as pattern recognition or data classification, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons.

2.2 Use of Artificial Neural Networks

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques. A trained neural network can be thought of as an "expert" in the category of information it has been given to analyze. This expert can then be used to provide projections given new situations of interest and answer "what if" questions. Other advantages include:

1. Adaptive learning: An ability to learn how to do tasks based on the data given for training or initial experience.
2. Self-Organization: An ANN can create its own organization or representation of the information it receives during learning time.
3. Real Time Operation: ANN computations may be carried out in parallel, and special hardware devices are being designed and manufactured which take advantage of this capability.
4. Fault Tolerance via Redundant Information Coding: Partial destruction of a network leads to the corresponding degradation of performance. However, some network capabilities may be retained even with major network damage.

2.3 From Human Neurons to Artificial Neurons

We conduct these neural networks by first trying to deduce the essential features of neurons and their interconnections. We then typically program a computer to simulate these features. However because our knowledge of neurons is incomplete and our computing power is limited, our models are necessarily gross idealizations of real networks of neurons.
Some interesting numbers

<table>
<thead>
<tr>
<th>BRAIN</th>
<th>PC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vprop=100m/s</td>
<td>Vprop=3*10^8 m/s</td>
</tr>
<tr>
<td>( \omega_n = 100h_z )</td>
<td>( \omega_n = 10^9 h_z )</td>
</tr>
<tr>
<td>N=10^{10}-10^{11} neurons</td>
<td>N=10^9</td>
</tr>
<tr>
<td>The parallelism degree ( \sim 10^{14} ) like ( 10^{14} ) processors with 100 Hz frequency. ( 10^8 ) connected at the same time.</td>
<td></td>
</tr>
</tbody>
</table>

1.4 Network Layers
- The commonest type of artificial neural network consists of three groups, or layers, of units: a layer of "input" units is connected to a layer of "hidden" units, which is connected to a layer of "output" units.
- The activity of the input units represents the raw information that is fed into the network.
- The activity of each hidden unit is determined by the activities of the input units and the weights on the connections between the input and the hidden units.
- The behavior of the output units depends on the activity of the hidden units and the weights between the hidden and output units.

3. DATA FILTERS

That is all we need to know about Neural Nets, if we want to make forecasts. The next problem with which you will have to fight - is noise in real signals. And the biggest trouble, that nobody knows what is noise and what is pure signal.

![Fig.1: Predictable novice in real signals](image)

3.1 Adaptive filters
Main features, which we must obtain:

1. Zero-phase distortion (because we are going to make forecasts and we shouldn’t influence on the data’s phase we have.
2. We don’t know what real signal is and what is noise, especially in case of share values or smth like that. So one have to decide apriori what is the signal we want to obtain.
3. In this article I’m going to make an overview of 3 types of adaptive filters: Zero-phase filter (smoothing filter), Kalman filter (error estimator), and Empirical mode decomposition.

3.2 Zero-phase filter
The main equation of this filter:

\[ y(n) = b(1)x(n) + b(2)x(n - 1) + \ldots + b(nb + 1)x(n - nb) - a(2)y(n - 1) - \ldots - a(na + 1)y(n - na) \]

The scheme of this filter

![Implementation of zero phase filter](image)

Where \( x(n) \) is the signal we have at the moment \( n \). \( Z^{-1} \) – means that we take the previous value of the signal (moment \( n-1 \)). \( b, a \) - are the filter coefficients.

Filter performs zero-phase digital filtering by processing the input data in both the forward and reverse directions. After filtering in the forward direction, it reverses the filtered sequence and runs it back through the filter. The resulting sequence has precisely zero-phase distortion and double the filter order. Filter minimizes start-up and ending transients by matching initial conditions, and works for both real and complex inputs. Note that filter should not be used with differentiator and Hilbert FIR filters, since the operation of these filters depends heavily on their phase response.

Problem: After the neural net the forecast will have 1 day delay if we tried to make forecast for one day. It looks like the picture on the right. But I should mention that all statistical values e.g. correlation coefficient are very good. It means that it’s is possible to use this filter, but with some kind of error estimator

3.3 Kalman filter (error estimator)
Consider a linear, discrete-time dynamical system. The concept of state is fundamental to this description. The state vector or simply state, denoted by \( x \), is defined as the minimal set of data that is sufficient to uniquely describe the unforced dynamical behavior of the system; the subscript \( n \) denotes discrete time. In other words, the state is the least amount of data on the past behavior of the system that is needed to predict its future behavior. Typically, the state \( x_k \) is unknown. To estimate it, we use a set of observed data, denoted by the vector \( y_k \). In mathematical terms, the block diagram embodies the
Following pair of equations:

The equation of this filter is given below:

\[ \hat{x}(n) = a \hat{x}(n-1) + k(n)[y(n) - ac \hat{x}(n-1)] \]
where \( x(n) = ax(n-1) + w(n-1) \) model of generating signal, 
\( w(n) \) - white noise and
\( y(n) = cx(n) + \nu(n) \) signal after neural net 
\( \nu(n) \) - white noise

Problem: After Kalman filter the error became less, but there is still a day delay.

The solution is Empirical Mode Decomposition.

3.4 Empirical Mode Decomposition

A new nonlinear technique, referred to as Empirical Mode Decomposition (EMD), has recently been pioneered by N.E. Huang et al. for adaptively representing nonstationary signals as sums of zero-mean AMFM components. Although it often proved remarkably effective, the technique is faced with the difficulty of being essentially defined by an algorithm, and therefore of not admitting an analytical formulation which would allow for a theoretical analysis and performance evaluation. The purpose of this paper is therefore to contribute experimentally to a better understanding of the method and to propose various improvements upon the original formulation. Some preliminary elements of experimental performance evaluation will also be provided for giving a flavor of the efficiency of the decomposition, as well as of the difficulty of its interpretation.

The starting point of the Empirical Mode Decomposition (EMD) is to consider oscillations in signals at a very local level. In fact, if we look at the evolution of a signal \( x(t) \) between two consecutive extrema (say, two minima occurring at times \( t^- \) and \( t^+ \)), we can heuristically define a (local) high-frequency part \( \{d(t), t^- \leq t \leq t^+\} \), or local detail, which corresponds to the oscillation terminating at the two minima and passing through the maximum which necessarily exists in between them. For the picture to be complete, one still has to identify the corresponding (local) low-frequency part \( m(t) \), or local trend, so that we have \( x(t) = m(t) + d(t) \) for \( t^- \leq t \leq t^+ \). Assuming that this is done in some proper way for all the oscillations composing the entire signal, the procedure can then be applied on the residual consisting of all local trends, and constitutive components of a signal.
can therefore be iteratively extracted. Given a signal $x(t)$, the effective algorithm of EMD can be summarized as follows:

1. Identity all extrema of $x(t)$
2. Interpolate between minima (resp. maxima), ending up with some envelope $e_{\text{min}}(t)$ (resp. $e_{\text{max}}(t)$)
3. Compute the mean $m(t) = (e_{\text{min}}(t) + e_{\text{max}}(t))/2$
4. Extract the detail $d(t) = x(t) − m(t)$
5. Iterate on the residual $m(t)$

In practice, the above procedure has to be refined by a sifting process which amounts to first iterating steps 1 to 4 upon the detail signal $d(t)$, until this latter can be considered as zero-mean according to some stopping criterion. Once this is achieved, the detail is referred to as an Intrinsic Mode Function (IMF), the corresponding residual is computed and step 5 applies. By construction, the number of extrema is decreased when going from one residual to the next, and the whole decomposition is guaranteed to be completed with a finite number of modes. Modes and residuals have been heuristically introduced on “spectral” arguments, but this must not be considered from a too narrow perspective. First, it is worth stressing the fact that, even in the case of harmonic oscillations, the high vs. low frequency discrimination mentioned above applies only locally and corresponds by no way to a pre-determined sub band filtering (as, e.g., in a wavelet transform). Selection of modes rather corresponds to an automatic and adaptive (signal-dependent) time-variant filtering.

Let’s try to do this procedure with next signal $x = 0.7 \sin(20\pi t) + 0.5 \sin(400\pi t), t \in [0,1]$
Let’s try to filter from high frequency part (light gray – EMD, dark gray – zero phase filters)

![Figure 5](image)

**Fig.5:** Filter from high frequency part (light gray-EMD, dark gray-zero phase filters)

Problem: Using this method we have strong boundary effects. Sifting process can minimize this effect, but not at all.

### 3.5 Zero-phase filter + Kalman filter + EMD = Solution

The so-called solution is next. We should use data, obtained from zero-phase filter on the boundaries, we should use linear weighed sum of zero-phase and some IMF’s in the middle. And after the forecast is done, we should apply Kalman filter to the forecast. Let’s try to do that. The figure on the bottom is the forecast of some company’s share value (production set). There is no delay.

![Figure 6](image)

**Fig. 6:** linear weighed sum of zero-phase and some IMF’s (forecast of some share values)

The bottom figures are zoomed pictures of previous figure

![Figure 7](image)

**Fig.7:** Both are zoom pictures of figure no 6
If you are going to make forecast, you will need filters. The best way to make forecast is to use adaptive filters. Three of them I showed you. The thing I will show next isn’t the best way to make forecasts. I’m going to teach ANN to make forecast for one day. After that we suggest predicted value as an input to make forecast for the second day and so on. Let’s try to make forecast for Siemens share values.

Here is Siemens function (for share values)

Let’s filter it.

Now we are ready to make forecast.

Here is the forecast in comparison with real data.
Of course it’s is better to use another techniques to make forecast’s for few days. But this approach can give you a chance to make forecast for 5-6 days depending on the function you are working with.

5. CONCLUSION

We had compared both of inspection method in industrial application. The real time inspection method can be expected to improve quality control in manufacturing environment. However, the implementation of computer vision in manufacturing is not simply image processing problem. The real time visual inspection is an integration system of lighting system, image acquisition, computer, controller and handling equipment. The results show that the dimension can be calculated using image processing algorithm which measures the length, width, edge and diameter of press part.

6. FUTURE WORK

The results show the potential of the system to be implemented in metal-based industry. It is shown that the above algorithm can be used to perform real time visual inspection system. The routine consists of digital image acquisition, noise reduction, edge detection, and feature extraction. However, future work is necessarily needed to inspect another parameter of press part such as straightness, flatness, roundness, angle, profile, and weight for part with more complex shape.

REFERENCES


