National conference on Engineering Innovations and Solutions (NCEIS – 2018)
International Journal of Scientific Research in Computer Science, Engineering and Information Technology
 | Volume 4 | Issue 6 | ISSN : 2456-3307

Absolute Magnetic Encoder Design Based On RBF Neural Networks

N. Sangeetha, Rajashekar J. S
EIE Department, Dayananda Sagar College of Engineering, Bangalore, Karnataka, India

ABSTRACT

This paper proposes absolute magnetic encoder design for analog angular measurement using multi-sensor data-fusion based on Radial Basis Function (RBF) neural networks. Multiple linear Hall effect sensors and a magnet are used to realize the analog angular output. RBF neural networks are used to approximate multi-dimensional nonlinear function between the sensor values and angular position of the magnet. The parameters of the RBF network are determined by supplying the data for multiple sensor values and the corresponding angular position of the magnet. Trained RBF neural network can be used to obtain the analog angle output for the given sensor inputs and it can be implemented using 8 or 16-bit microcontroller. This design of the encoder allows flexibility in terms of placement of the sensors.
Keywords: Rotary encoder, magnetic encoder, multisensor data fusion, hall effect sensor, ANN, RBF neural networks, analog angular measurement.

I. INTRODUCTION

Rotary encoders are often used to track the angular position of the motor shaft. These are commonly used in CNC machines, robots, and other industrial equipment. Rotary encoders, whether absolute or incremental, typically use one of two measuring principles — optical or magnetic. While optical encoders were, in the past, the primary choice for high resolution applications, improvements in magnetic encoder technology now allow them to achieve better resolution. Magnetic technology is also, in many ways, more robust, rugged, having excellent shock resistance, fast, durable to unclean environment, reliable at low temperatures and immune to dust and dirt than optical technology, making magnetic encoders a popular choice in industrial environments.

An absolute rotary magnetic encoder design with multi-sensor data fusion based on radial basis function artificial neural networks is proposed for the analog angular measurement. This design makes use of multiple linear hall-effect sensors positioned at various locations around the rotating magnet.

Artificial Neural Networks (ANNs) are computing systems inspired by biological neural networks. ANNs learn by considering examples and they don't generally need task specific programming. An ANN consists of connected units called artificial neurons, which are simplified version of biological neuron. Each connection between artificial neuron can transmit signal. In common ANN implementations, signal is a real number. The output of each artificial neuron is calculated by applying nonlinear function of the sum of its inputs. Typically, ANNs are organized in layers. Different layers may perform different kinds of transformations on their inputs.

Each connection in ANN has a real value associated with it called weight that represents the signal strength. The weight values associated with connections in ANN goes on adjusting as the learning proceeds. ANNs have been successfully applied on variety of tasks such as image recognition, speech recognition, medical diagnosis etc.

Broadly speaking, ANNs can be applied for two major categories of tasks: 1. Classification 2. Function approximation. For magnetic encoder, a special type of ANN called Radial Basis Function (RBF) neural networks are used for approximating the multi-dimensional nonlinear function between sensor outputs and the angle value.

RBF network is an ANN that uses radial basis functions as activation functions. RBF networks usually consists of single hidden layer. The number of neurons in hidden layer is a design parameter. The output of the network is a linear combination of radial basis functions of the inputs and weights of the network. RBF networks have many uses such as function approximation, time series prediction, classification and system control.

The common layered ANNs use back-propagation algorithm for learning. The learning process for RBF ANNs is slightly different compared to common ANNs. As the learning proceeds, the weights in RBF network keeps on adjusting so as to output a continuous value that approximates some nonlinear function.

This design of magnetic encoder with RBF networks results in reduced hardware complexity because the sensors need not be positioned at some predetermined accurate locations. Also, the results for angle output value showed less amount of standard deviation.

The rest of this paper is arranged as follows. Section II describes Magnetic encoder architecture, RBF network architecture along with its training. In section III the trained RBF network output and its accuracy are discussed. Section IV provides the conclusion.

II. METHODOLOGY

A. Magnetic encoder architecture

Figure 1 depicts the proposed architecture for magnetic encoder. It consists of multiple linear hall-effect sensors placed around a rotating magnet. As the magnet rotates, the output from the linear hall-effect sensors varies. Minimum 3 sensors are required to uniquely identify the angular position of the magnet.

The output signals from all the sensors are sent to a microcontroller that implements RBF neural network. For training the RBF network, sensor values for various angular positions of the rotating magnet needs to be sent to the network trainer. The trainer determines the weights of the RBF network so as to approximate the multi-dimensional nonlinear function between sensor outputs and angle value with desired level of accuracy.

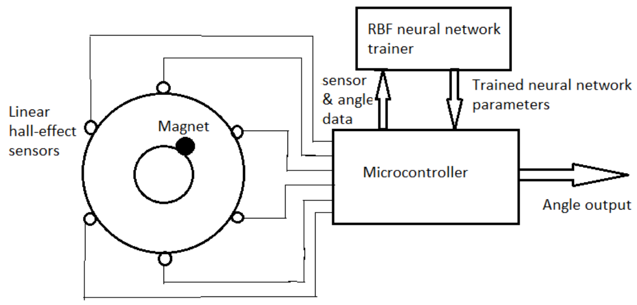

Figure 1. Magnetic encoder architecture

After completion of the training, all the weights of RBF network are sent to microcontroller. These weights are stored in nonvolatile memory of the microcontroller. With the trained weights, the microcontroller can output the analog angle value for the sensor inputs. The training needs to be performed in a machine with high computing capabilities as it requires larger memory, higher bit resolution and faster CPU. Training needs to be done only once in the beginning for a given configuration of multiple sensors and magnet.

B. RBF network architecture

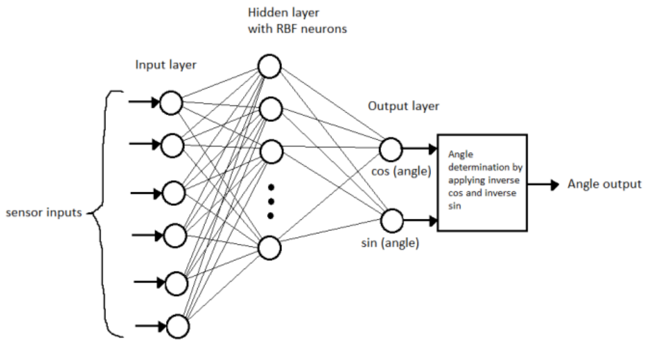

Figure 2. RBF network architecture

The above figure depicts the structure of the RBF network consisting of 3 layers:

1. Input layer consisting of nodes for each sensor.
2. Hidden layer consisting of RBF neurons.
3. Output layer with 2 nodes for cos(angle) and sin(angle).

The number of nodes in hidden layer is a design parameter of the network. For the magnetic encoder implementation, 5 to 10 RBF nodes in hidden layer are found to be sufficient. The nodes in hidden layer uses Gaussian activation function as follows:

$$e^{\left(\frac{-\beta(x-c)^2}{r^2}\right)}$$

where c is the center of the RBF neuron
r is radius associated with RBF neuron
β is parameter for controlling shape of the activation function.
x is the input i.e. multiple sensor values in this case.

Each RBF neuron has its own c, r values. $\beta = 1$ is chosen for all the RBF neurons.
The centers of the RBF neurons are chosen at equal angle intervals. And radius value chosen is the Euclidian distance between these centers.

There are no weights associated with connections from input layer neurons to RBF neurons. Weights are associated with connections between hidden layer neurons and output layer neurons.

Output layer consists of two nodes corresponding to cos(angle) and sin(angle) values. The cos and sin outputs are chosen instead of single angle value as output because of the cyclical nature of the inputs and output angle values. Both these output nodes are linear activation neurons.

One set of weights are associated with output neuron for cos(angle), second set of weights are associated with output neuron for sin(angle). The values for these weights associated with both the output neurons needs to be determined by training the RBF neural network. For training, the values for multiple sensor inputs and corresponding cos(angle) and sin(angle) values needs to be supplied to the network trainer. The training methodology is described in next section.

C. RBF network training

Let A denote a matrix with number of columns = number of sensors and number of rows = number of data points.
Let B denote a two-column matrix with values for cos(angle) and sin(angle) corresponding to the angle position for each row from matrix A.
Let W denote a two-column matrix representing weights from hidden layer to output layer. First column in this matrix consists of weights associated with cos(angle) output neuron and second column consists of weights associated with sin(angle) output neuron.
Since the output layer consists of linear activation neurons, the relationship can be expressed in matrix form as follows:

A W = B

The weights or the elements in W matrix are the unknowns. The number of these weights depends upon the number of RBF neurons in the network. And the number of equations is equal to the number of training data available, which is usually much more than the number of RBF nodes. The more the number of training data available, better will be the accuracy of approximating the function. From linear algebra, the unknown weights can be expressed as follows:

$W = (A^T A)^{-1} A^T B$

III. RESULTS

The trained network outputs the cos(angle) and sin(angle) values for any given sensor inputs. The actual angle value can be obtained by applying inverse cos and inverse sin functions. Since the network consists of only one hidden layer and relatively less number of total neurons and weights in it, its response in terms of time for obtaining the angle output value will be relatively fast. Also, the memory requirements are not huge and it can be implemented using 8 or 16-bit microcontroller. However, for training the network, machine with high computational capabilities are required as it requires high memory, faster CPU and higher bit resolution.

The sensor outputs, expected angle value and actual angle value obtained are shown in Figure 3. Note that the expected and actual angle values are overlapping in below figure because of less difference between them.

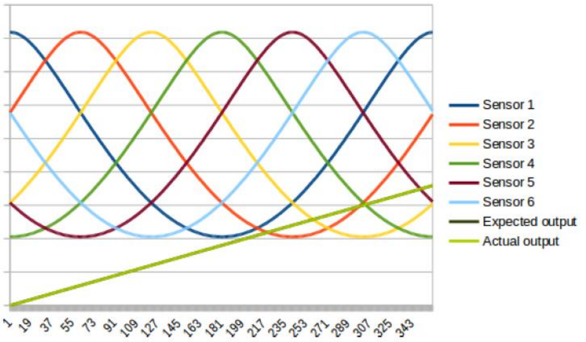

Figure 3. Angle output of RBF network

IV. CONCLUSION

This paper shows an alternative approach of analog magnetic encoder design based on RBF neural networks. With this design, accuracy of ± 0.6 degrees is obtained. Future work can be carried out to improve the accuracy further, reduce memory requirements by having less RBF network parameters, improve the frequency response.

V. REFERENCES

[1] H. Lee, B. Lee, K. Park, and R. Elmasri, "Fusion Techniques for Reliable Information: A Survey," International Journal of Digital Content Technology and its Applications, vol. 4, no. 2, pp. 74–88, 2010.

[2] H. T. Le, H. V. Hoang, and J. W. Jeon, "Efficient method for correction and interpolation signal of magnetic encoders," in Industrial Informatics, 2008. 6th IEEE International Conference on, 2008, pp. 1383–1388.

[3] Arthur Noronha Montanari and Everthon de Souza Oliveira, "A Novel Analog Multisensor Design Based on Fuzzy Logic: A Magnetic Encoder Application", IEEE Sensors Journal, DOI 10.1109/JSEN.2017.2752959.

[4] J.-S. Jang and Chuen-Tsai Sun, "Neuro-fuzzy modeling and control," Proceedings of the IEEE, vol. 83, no. 3, pp. 378–406, 1995.

[5] https://en.wikipedia.org/wiki/Artificial_neural_network

[6] https://en.wikipedia.org/wiki/Radial_basis_function_network