A Distributive Tactile Sensor to Determine an Applied Load Position Using Different Neural Network Training Strategies

เพ็ญศิริ ทองผดุงโรจน์
ภาควิชาวิศวกรรมการผลิต คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ
1518 ถ.พิบูลสงคราม เขตบางซื่อ กรุงเทพฯ 10800
โทร 0 2913 2500 ต่อ 8208 โทรสาร 0 2587 0029 E-mail: pensiri@kmitnb.ac.th

Pensiri Tongpadungrod
Department of Production Engineering, Faculty of Engineering, King Mongkut's Institute of Technology North Bangkok
1518 Pibulsongkram Rd, Bangsue Bangkok 10800 Thailand
Tel: 0 2913 2500 Ext. 8208 Fax: 0 2587 0029 E-mail: pensiri@kmitnb.ac.th

Abstract

This paper describes an arrangement of a one-dimensional distributive tactile sensing system that can be used to determine a simple contact parameter of an applied position of a constant magnitude point load. The load position was determined using a back propagation neural network as an interpretation algorithm. Three neural network training strategies were attempted and reported in this paper. It was found that randomly chosen training positions resulted in the largest determination errors between 4.7 – 16.8% of the total beam length. The error was reduced to 4.6% when the training positions were at an equal pitch. It was also found that higher errors were obtained when the load position was close to either end of the beam because the beam deflection was of different sensitivity. To reduce this type of errors, cascade neural networks were introduced. The principal of the cascade approach relies on a crude filtering of load position in to a pre-defined section of the beam and fine-tuning the result to obtain a more precise position of load. The result of cascade neural network training strategy showed a reduction of determination error to 1.0%.

1. Introduction

A tactile sensing device has been defined as a device that is used to determine a contact interaction between the device and some stimuli [1]. In many applications tactile sensing has been devised to emulate the vision role in determining shape, size and position of a contacting object [2]. Such devices often employ a large number of sensing elements to ensure sufficient spatial resolution as each element is activated independently. This is sometimes referred to as 'discrete' sensing scheme. However, the discrete tactile sensing is achieved at a cost of complex construction and high computational load. In contrast to the discrete scheme described above, the approach described in this paper is able to discriminate between different loadings and offers the potential to minimise the number of data sets [3], [4]. This method is referred to as 'distributive' because it relies on a continuum medium of which responses to a contact can be detected entirely over the active area rather than at a local contact point. The sensing elements of a distributive system are not necessary an integral part of the surface and the measurement points can be appropriately selected across the surface area. The determination of contact is made from the unique patterns of the sensing information that vary between contact types.

2. Arrangement of the test rig

A distributive system used as the case study has been defined as a simply supported beam structure. An applied load position on the beam is to be determined from the deflection. The applied load has been defined as a case of a constant point load. The chosen system has advantages of simplicity where its performance can be easily defined and extended to more complicated cases such as a two-dimensional system. In terms of applications, the case study has an important implication in force sensing such as force-feedback tools. An example of such tools is an endoscope which has a force-feedback sensing unit installed at its tip. The arrangement of the system under study is shown in Figure 1.

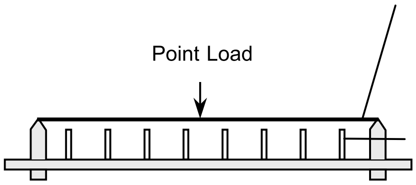

Figure 1. The one-dimensional distributive beam system under study

The one-dimensional surface is a mild steel beam of dimensions 40 × 400 × 1.2 mm supported at both ends by wedges. A 3 N load having a point contact with the surface is applied on the surface and the induced deflection measured at 8 positions spaced at an equal pitch of approximately 44 mm. The deflection of the beam at the measurement points can be obtained experimentally using transducers such as proximity infrared sensors [1] as well as by simulation.

3. The simulation of the surface deflection

In order to efficiently study performances of the described system, a mathematical model describing the behaviour of the system has been defined and the corresponding computational simulation developed.

The deflection behaviour of the beam surface can be calculated from the standard beam bending theory reported in most structural mechanics texts for example [5]. For a simply supported beam, the deflection y at position x in response to an applied load W at position a on a simply supported beam of length l is given by:

$$y = \frac{W}{6EI}\left[\frac{(2l-a)(a-l)ax + (l-a)x^3}{l} - < x - a >^3\right] \quad (1)$$

A computer program used to simulate the beam deflection was developed. The simulated deflection and the measurements were found to be in good agreement. An example comparison between simulated and measured deflections is shown in Figure 2, where a load of 3 N was applied at the centre of the beam.

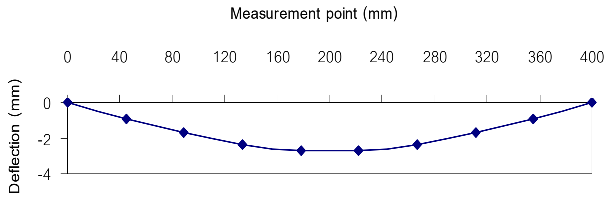

Figure 2. Simulated beam deflections

4. The determination of an applied load position

The determination of an applied load position was carried out using a neural network as an interpretation algorithm. The neural network takes the beam deflection at measuring points as inputs and interprets for the corresponding applied load position as the output. Prior to the process of interpretation, the neural network must be trained to establish the relationship between the input and the output from selected load positions that will be referred to as the training positions. This section investigates the performance of the neural network to determine an applied load position for the networks trained with random and equally spaced positions.

4.1 Training procedure

In the training process, load positions and the corresponding beam deflections at points of measurement were obtained for training. As we wish to determine an applied load position from measured deflection, the former was the network output and the latter the neural network input. The number of input data was fixed at 8 points of measurement. For each neural network, 10 load positions were given for training. The training was carried out to achieve the error between the desired and the network outputs of no more than 0.001%. The momentum rate and the learning rate were fixed at 0.9 and 0.7 respectively. The networks created received 8 inputs (beam deflection) with one hidden later and 10 hidden nodes and outputs a single value of the load position.

4.2 The performance of the network to determine an applied load position

After the neural network was trained to achieve the specified conditions, it was used to determine an applied load positions from deflection at the points of measurement. The performance of each network was obtained using test data of 399 load positions at an equal pitch of 1 mm. It should be noted that some of the test data maybe the same as the training positions, but most were not.

In some of the analyses carried out following this section, the performance was obtained in terms of determination positional error that was calculated from [6]

$$e = \frac{|A - P|}{l} \times 100\% \qquad (2)$$

Where e is the positional error of the beam length (%), A is the applied load position (mm), P is the determined load position (mm), and l is the beam length (mm).

4.3 Random training positions

In the first investigation, random load positions were used to train the neural network. In order to see the overall trend of the accuracy of the system, 10 sets of training positions were randomly generated and the corresponding induced deflections at measurement points obtained using equation (1). Each set was used to train the neural network to meet the specified conditions. The test of performance to determine an applied load position was carried out using the test data described in 4.2. The positional errors of each networks were calculated using equation (2) and the average value plotted in Figure 3.

Figure 3 Average positional errors of the networks with random training positions

Figure 3 shows that the average positional error of the determination of an applied load position varies greatly between 4.7 – 16.8% of the beam length when the training positions were randomly selected. On average the networks trained with random load positions was considerably high at 11.1% (corresponding to approximately 44.4 mm). It is not surprising that the randomly selected training positions resulted in such diverse accuracy and tendency of high errors. The reason for this is that the random positions chosen for training can be more clustered in area than others. Because of clustering, the network was likely biased towards to the area of high density of training positions and produced higher areas where there were less data available in training.

4.4 Equally spaced training positions

Solving the clustering effect of the random training positions can be conveniently achieved by choosing the training positions such that they well cover the entire beam length. To achieve this, the training positions were chosen with a uniform distribution across the beam length with an equal separation of 40 mm. The first training position was chosen at 20 mm from the left end and the rest of the training positions were located at 40 mm separation from one another. After training with the described equally pitched positions, the network was tested for its performance using the test data described in 4.2. The determination of an applied load position is plotted in Figure 4.

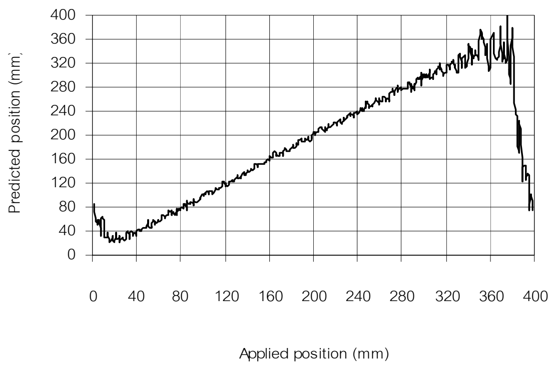

Figure 4 The determination of an applied load position with training positions at an equal pitch

On average the positional error was 4.6% of the beam length. This was comparable to the smallest error obtained when the training positions were random (training number 8 in Figure 3) but significantly smaller than other sets of random training positions. As expected, the equally uniform training positions are more effective than random positions.

The plot in Figure 4 shows that high errors were obtained when load was applied near either end of the beam. This was due to the variation in sensitivity of the beam's response to an applied load. When load was applied near the ends, the induced deflections were of small magnitudes from which the network determined an applied load position with a greater difficulty resulting in high errors. In contrast, load applied near the beam centre induced larger deflections that were more easily captured. Despite of this phenomenon, the network trained with 10 load positions uniformly distributed did not take into account the difference in the sensitivity of the inputs. Although it maybe suitable to use a single network to determine an applied position when load application is concentrated near the centre of the beam, it may not be the best solution when load application is across the entire beam length.

5. Cascade neural network

The cascade neural network has been implemented to respond to different types of neural network information. For example in [7] the cascade approach was used to determine different parameters such as position, width and magnitude of load on a one-dimensional distributive tactile sensing surface. In this paper, the cascade network will be used to determine load positions in different sections of the beam.

5.1 The architecture of cascade neural network

As can be seen in Figure 4 in 4.4, the accuracy in determining an applied load position across the beam length can be divided into 3 sections that are on the left and right end of the beam and in the centre. The criteria on which how the section should be divided is based on the work by Tongpadungrod [1] which suggested that the sensitivity of the beam length at each quarter of the beam length at either end was different from the rest of the beam. By using such criteria the beam can be divided into 3 sections at 0 –100 mm (section 1), 100 – 300 mm (section 2) and 300 – 400 mm (section 3). To use the cascade networks to respond to each section of the beam separately, it is essential to firstly classify the load position into which the section of the beam it was applied. This can be viewed as a simple filter to match the load position into a sub-section before fining tuning for a more precise result. The architecture of the cascade networks is shown in Figure 5.

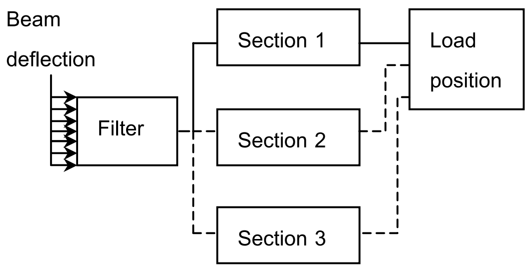

Figure 5 Architecture of the cascade neural network

The cascade network used in this approach comprises 4 sub-networks, one of which for section classification and the others for precise determination of an applied load position.

5.2 Classification of an applied load position

In the first classification, the beam deflection was passed to a filtering network to determine the section of the beam in which the load was applied. In general one will find that the neural network is of high accuracy when used as a classification tool. A similar work regarding this aspect of neural network usage can be found in [1].

Like previous use of neural network, the neural network for section classification was trained with selected load positions. Because in 4.4 it was shown that the equally spaced training positions were most effective, they were chosen to train the networks in this part of study. In order to provide sufficient training data, 10 applied load positions at an equal pitch in each section discussed in 5.1 were picked for training the networks. Note that sharp boundaries were applied when determining section of an applied load. For the section filtering network, the network input was of 8 components (beam deflections) and the output comprised 3 components corresponding to the specified sections of the beam. The hidden layer between the input and the output comprised 10 hidden nodes. The momentum rate and learning rate were 0.9 and 0.7 respectively. Because determination of an applied section was a coarse filtering process, the error between the desired and the network outputs was higher than that used for precise determination of an applied load position at 0.1%.

The trained network for classifying section of an applied load was tested for its performance using the set of load positions at an interval of 1 mm. It was found that most positions tested were classified into the correct section. The sectional error was satisfactorily low at 0.5% of all points tested.

5.3 The determination of an applied load position from the sectional classification

This part of the paper is the fine-tuning process to determine a more precise position of an applied load by taking the result in 5.2 and passing it to the corresponding sectional network to determine the load position as a continuous function as in 4.4.

The sub- networks corresponding to each load application in each section of the beam were trained using 10 equally distributed applied positions in each section of the beam. Training parameters of all three networks were fixed at the same values as follows. The momentum rate and learning rate were at 0.9 and 0.7 respectively. The hidden layer comprised 10 hidden nodes and the error between the desired and network output (that was the load position) was no more than 0.001%.

The trained networks were tested by passing the data obtained in 5.2 to the corresponding network. In total 399 load positions were tested and the result is shown in Figure 6.

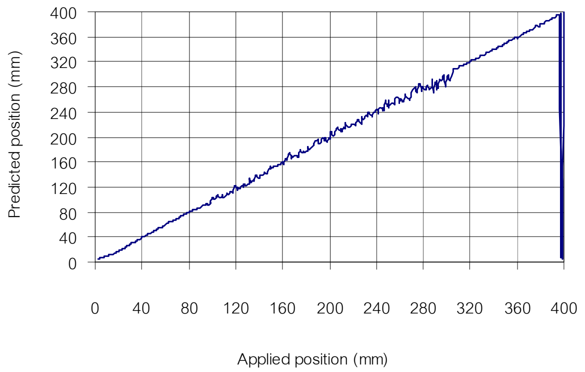

Figure 6 Determination of an applied load position using cascade neural network

The result in Figure 6 shows that the errors at ends of the beam were significantly reduced compared to the result obtained when a single network was used to determine an applied load position (see Figure 4). However, the cascade networks did not help reducing the errors in the middle section of the beam.

The average error obtained in the cascade approach was 1.0%. This shows a significant reduction in error compared to when a single network was used where the average error of the entire beam length was 4.6%. However, by observation it was found that there were 2 odd errors in the determination of the section of load position before the result was cascaded to the fine-tuning process and resulted in unexceptionally high errors of over 97%. With the exclusion of the unusual errors discussed, the average determination error of the cascade networks was reduced to 0.6% of the beam length. The effect of odd sectional errors could be reduced by applying a fuzzy algorithm for applied positions closed to the boundaries between sub-sections.

6. Conclusions

The distributive tactile sensing system is an effective tool for identifying load parameters offering a reduction in the number of sensing elements and construction complexity. The work presented is a one-dimensional surface on which load position can be determined using the neural network algorithm. It was found that the accuracy in determining an applied load position is affected by the neural network training strategies. It was found the sensitivity of the system was the lowest when the training positions were selected on a random basis as can be seen from the determination error at an average of 11.1% for 10 networks. With equally pitched training positions, the error was reduced to 4.6% of the beam length. It was found that determination of load positions in approximately either quarter of the beam length near the ends resulted in higher errors compared to when load was applied near the beam centre. In order to moderate such errors, the cascade neural network was introduced to determine load applied near the ends and the middle section of the beam separately. It was found that the cascade neural network was the most effective training strategy as the determination error was the lowest at 1.0%.

References

[1] Tongpadungrod, P., “Characteristics of distributive tactile sensing system”, PhD thesis, University of Bristol, 2002

[2] Holweg, E.G.M. and Jongkind, W., “Object recognition using a tactile matrix sensor”, EURISCON’ 94 Advanced Manufacturing and Automation RES 1994, Malaga, Spain, 1379-1383, 1994

[3] Brett, P.N. and Li, Z., “A tactile sensing surface for artificial neural network based automatic recognition of the contact force position”, Proc Instn Mech Engrs, 214(I), 207-215, 2000

[4] Dargahi, J., Parameswaran, M., and Payandeh, S., “A micromachined piezoelectric tactile sensor for an endoscopic grasper : Theory, fabrication and experiments”, Journal of microelectromechanical systems, 9(3), 329-335, 2000

[5] Young, W.C., “Roark’s formulas for stress and strain (International Edition)”, McGraw-Hill Inc, Singapore, 1989

[6] Tongpadungrod, P., Rhys, T.D.L., and Brett, P.N., “An approach to optimise the critical sensor locations in one-dimensional novel distributive tactile surface to maximise performance”, Sensors and Actuators A, 1005, 47 – 54, 2003

[5] Fearing, R.S., “Tactile sensing mechanism”, International Journal of Robotics Research, 9(3), 3-23, 1990

[6] Gomm, J.B., Weerasinghe, M., and Williams, D., “Diagnosis of process faults with neural networks and principal component analysis”, Proceedings of the Institution of Mechanical Engineers. Part E, Journal of process mechanical engineering, 214(2), 131-143, 2000

[7] Ma, X. and Brett, P.N., “A distributive 1 dimensional tactile surface for determining the position, width and intensity of a distributive load”, IEEE Transactions on Instrumentation and Measurements, 51(2), 331 – 336, 2002