Application of Object Recognition in Locomotive Components Monitoring

Liming XIE, Kai YANG, Jianping PENG, Xiaorong GAO, You TAN

Southwest Jiaotong University, Chengdu, China

Contact e-mail: m15982309732@163.com

Abstract. The operation safety of locomotive is related to every components on the locomotive. So, understanding the running state of every components timely is necessary. And the most difficult in locomotive key components monitoring is how to define if the components is abnormal due to the judgment is subjective. Such as, the same shift appear on chain or flexible pipe and on fixed bolt is completely different. In this paper, an improved Hierarchical Temporal Memory network was presented, and the purpose of the improvement is focused on the construction of the network. What’s more, the adaptability for illumination and translation was considered. Result in this paper show that it is feasible to improve the accuracy of abnormality warning.

Introduction

Anomaly inspection of the running gear is the most important work in train safety inspection. Cause the running condition of running gear is closely related to the safety of train operation. It is not just a very important work but also a very intricate due the variety and huge numbers of trains. It is not very reliable for doing the work by human absolutely, because it is a very fussy work. So an automatic inspection system of the running gear is urgent. The most difficult in automatic inspection of running gear is how to define the anomaly. It is almost subjective for the criterion of anomaly. For example, the sifting of flexible pipe and bolt are totally different. So, making a difference between the objects is necessary. The problem became an object localization task.

Object recognition is a very important task in a wide variety of applications. Humans can recognize a considerable number of objects with little effort. Therefore, a significant amount of work has been done in this field by a large number of researchers. Although computer vision techniques provide adequate solutions to the problem of object recognition, achieving human-like capabilities is still a long distance away.

There are two main questions on image recognition that need to be answered: (1) How do we represent objects? (2) How do we classify objects?

In this paper, we present an image recognition system based on the hierarchical temporal memory (HTM) algorithm. The main idea is to structure the system on the basis of the prediction theory. The HTM algorithm is a biologically inspired framework proposed by Hawkins and George (2008, 2005, 2009) as the first practical implementation of the memory-prediction theory of the brain function presented by Hawkins and Blakeslee (2004). A hierarchical structure network and a sparse representation are used in this theory. This
A hierarchically structured network is used for simulating the ability of abstraction, and the sparse representation is used for simulating the visual sensing cells. A considerable amount of work based on HTM has been proposed by researchers in the past few years. Csapó et al. (2007) introduced a complicated algorithm for distinguishing objects presented in different scales. Melis and Kameyama (2009) implemented the HTM algorithm for traffic sign recognition. Further, Bundzel and Hashimoto (2010) introduced a system for exploring an environment and recognizing different types of objects in it. Therefore, it is evident that the use of HTM in an object recognition problem can significantly improve the recognition performance.

In this paper, a hierarchically structured machine learning technique is proposed on the basis of the HTM theory. Different from the mature HTM system, the network proposed in this paper focuses on static image recognition. Moreover, the proposed system does not focus on intelligent learning. The proposed network also has a temporal module for feature selection and a spatial module for object verification.

1. Hierarchical temporal memory structure

The HTM network is organized in layers of elementary units called nodes. All the nodes implement the same learning and inference algorithms, and therefore, they operate in computationally identical regimes. They only differ in the content of their internal memory, i.e., the information gained during the learning phase. The individual layers of the nodes are usually structured into a tree-like hierarchy. There is always a zero sensory level that serves as an input to the first level of nodes. Further, a single node called ZetaTopNode is typically situated at the top of network. It performs the task of a simple classifier that associates the learned top-level belief patterns with the known output categories (Numenta, 2007).

Fig. 1. Structure of the HTM network for learning invariant representations of binary line drawings. This image is derived from a figure in (George, 2008)

A simple example of an HTM network is shown in Fig. 1. Each frame of the training sequences is presented to a sensory field (“retina”) measuring 32 × 32 pixels. The nodes at level 1 are arranged in an 8 × 8 grid where each node receives an input from a 4 × 4 pixel retina patch. Such an arrangement of level-1 nodes covers the entire retina with no overlap between neighboring patches. The effective input area (patch) from which a node receives its input is called the field of view or the receptive field. At level 2, each node receives its input from a 2 × 2 patch of the child nodes at level 1 with no overlap. The single node at the
top-most level, level 3, is connected to all 4 × 4 nodes at level 2 and thus, covers the entire network’s input by pooling outputs from all its child nodes. Each node follows the same algorithmic procedure irrespective of the level that it belongs to (George and Jaros, 2007). This procedure involves two different modules, namely the spatial module and the temporal module, and has two distinct operation mechanisms: the first one is the training mode, which includes the spatial and the temporal procedures, and the second is the inference mechanism, where a node produces outputs to be fed into higher nodes.

1.1 Spatial module

There is an input area for every node, and the status of all the nodes in this area can be seen as a pattern. The processing of the network identify the input patterns can be treated as a spatial module. Due all the messages involved appeared in the same time and the patterns are related to their spatial place.

The principle behind the processing of learning is that every node must be used for representing some patterns. If a node is not activated after some loops, the connection coefficient between the node and its child nodes will change as the input.

1.2 Temporal module

While the nodes receive inputs from the child level in the spatial module, the temporal module obtains this output and starts learning. The primary task of the temporal module is to make predictions. The outputs of a node at times t and t+1 are connected, and depending on this connection, the node can predict the next pattern inputted by its child nodes.

All the nodes in the HTM network operate on the basis of the principle discussed above. After training, the network can be used for recognition and prediction, particularly those of temporal sequence patterns.

2. Optimization network

Image recognition is a spatial recognition task in the HTM network. However, it is very difficult for the network to fit multiple-object recognition because of the classification method used. In our approach, the structure of the HTM network is modified for the multiple-object recognition. The spatial module is replaced by a prediction space, and the temporal module is replaced by the “memory–prediction–judgment” processing. Moreover, in order to fit the multiple-object recognition, the structure of the network is modified as shown in Fig. 2.

![Fig. 2. Difference between traditional HTM network and modified HTM network used in our approach](image)

The key difference between the two networks is the object classification. In a traditional network, the last step of the recognition is the design of a classifier to classify the sub-features captured by the sub-level network; in contrast, in the proposed approach, the last step is the verification of whether the prediction based on the sub-features is correct.
2.1. Convolution features

The vision system works as an efficient and reasonable image processing system. Further, some cells between the retina and the cortex work as a filter that is sensitive to different inputs from the retina cells (Hubel and Wiesel, 1959, 1962, 1968). In our approach, we treat the cells as sensitive nodes between the input image and the feature map level. Further, convolution is used in this approach.

![Diagram of sensitive nodes](image)

In order to simulate the function of the retina cells, Barlow (1961) and Willshaw and Buneman (1969) proposed a method called sparse representation. The sparsity of an image can be described as follows:

\[
\min \|X\|_0 \quad \text{s.t.} \quad \|Y - DX\|_F^2 \leq \epsilon
\]

Where \(X\) denotes a sparse matrix and \(Y\) represents a sample. \(D\) denotes the basic vector called dictionary.

The sparse representation is used for acquiring the feature points of interest in this study. Therefore, the anti-interference ability of the dictionary is the most important characteristic considered.

![Feature map captured by using conventional network](image)

The dictionary captured by the sparse representation is worked upon as a conventional kernel in the network. The last step is to obtain the largest direction of every image pitch. Therefore, a feature map is captured by the network, as shown in Fig. 4.
It is very difficult for the network to remember the large number of object points represented in the feature map. However, fortunately, there is no need for the network to remember all the points of an object. It is feasible for the network to recognize the object by using several points of interest. A method for the selection of these points is required for the proposed network. In the proposed method, we select the maximum point in the convolution map of every kernel.

2.2 Modified spatial module

The spatial module is used for recognizing the input pattern in the HTM network. Further, the spatial module works as a tree-like network. This implies that the nodes in the upper level are activated by the active child-level nodes, as shown in Fig. 2(a). It is very difficult for the network to fit the translation of the object because the input vector of every upper node changes with the translation.

The function of the spatial module in our approach is almost the same as that in the HTM network. The only difference is that the former considers the center point of the patterns. This center point is derived from the convolution features. Thus, the function of the spatial module can be described as follows:

\[ Pre = \sum_{i=1}^{N} pre_i \]

Where \( Pre \) denotes the most possible pattern of the input and \( pre_i \) represents the prediction of every active child node.

In order to realize the function of the spatial module, a prediction space is used in the proposed method. The prediction space is an N-dimensional space, where \( N \) denotes the dimensions of features. The first step is to train the space, and therefore, we need to feed the space with more patterns. The corresponding network is shown in Fig. 5.

As shown in Fig. 5, the output of the spatial module is a series of possible patterns, and every pattern has a degree of confidence. Therefore, if the spatial module could speak, it would say, “I think the possibility of ‘A’ is 80%.”

There are two types of child nodes for one pattern. One is the wakeup node (the yellow node in Fig. 5), and the other is the active nodes (red nodes in Fig. 5). The wakeup node is at the center of the pattern, and the active nodes are the nodes around the wakeup node. During identification, the network moves the center point node by node (the nodes are activated) and searches the active nodes in the around by some principle. And make the prediction by the nodes captured.
Moreover, distortion must be considered in the spatial module. A concept named effecting area was introduced in this paper. It states that if a child node is activated, the nodes around the active node will be activated by it. Further, the node activity decreases with an increase in the distance between the nodes. Here, we use the Gaussian distribution to describe the decrease.

2.3 Modified temporal module

The function of the temporal module in the HTM network is the structural relationship between patterns that appear at different points of time. A considerable adjustment is required in the proposed network because the purpose of this study is image recognition. The network does not need to remember the patterns that appear at different points of time. Further, the most important task for the network is to identify the patterns as soon as possible.

The temporal module in the proposed network is used for remembering the relationship between the local patterns and the upper-level patterns. The relationship is used for activating the upper-level patterns by the local patterns. The network is formatted, as shown in Fig. 6.

![Diagram of the proposed network]

Fig. 6. Structure of the proposed network. There are four levels (except the input level) in the network, and every level works as a spatial module.

The number of levels in the network is not constant. In the same as HTM, the complexity of every level is connected to how to design the network. Further, the larger the prediction space is, the more complicated is the pattern that it can identify. If the prediction space is sufficiently large, the number of levels is small, and vice versa.

3 Experiment validation

In this section, the proposed method is evaluated. The test image is taken from the bottom of a locomotive. The aim of the network is to recognize the objects located on the locomotive, such as the break pad, drawing stick, and axis box.

In order to test the function of the proposed network, we design a four-level network. The first level is the input image, and the size of this image is not constant. The second level contains the convolution features, which are captured by eight $15 \times 15$ kernels. The kernels
are obtained by using a sparse representation, and the training samples are taken from the image edges. The third level is the level of the local features; here, these features are statistically obtained by using 10 convolution features. A 201 × 201 × 64 prediction space is used in the third level. The last level is the object level. The prediction space of level 4 is replaced by the structural maps of the objects. Note that the structural map of every node also fits the Gaussian distribution.

### Table 1. Recognition rate for the locomotive objects

<table>
<thead>
<tr>
<th>Locomotive type</th>
<th>Number of locomotives</th>
<th>Object types</th>
<th>Number of objects in one image</th>
<th>Number of recognized objects</th>
<th>Recognition rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>HXD1C</td>
<td>9</td>
<td>27</td>
<td>34</td>
<td>254</td>
<td>83%</td>
</tr>
<tr>
<td>HXD1D</td>
<td>15</td>
<td>22</td>
<td>42</td>
<td>593</td>
<td>94%</td>
</tr>
<tr>
<td>HXD3C</td>
<td>32</td>
<td>24</td>
<td>38</td>
<td>1069</td>
<td>88%</td>
</tr>
<tr>
<td>HXD3D</td>
<td>18</td>
<td>35</td>
<td>60</td>
<td>1039</td>
<td>96%</td>
</tr>
<tr>
<td>Total</td>
<td>74</td>
<td>108</td>
<td>3232</td>
<td>2955</td>
<td>91.4%</td>
</tr>
</tbody>
</table>

![Fig. 7. Recognized image (recognized objects are denoted in the yellow boxes)](image)

![Objects with higher recognition rate (100%)](image)

![Objects with lower recognition rate](image)

We train the network by using 108 objects from different types of locomotives and use only one image per object to identify the objects in the images of the locomotive bottom after training. The results are presented in Table 1, and a part of a recognized image is shown in Fig. 7.

Note that for HXD1C and HXD3C model, lower recognition rate was counted. It is caused by the features extraction approaches. Due the features extraction method is sensitive to the complexity objects. Some objects has higher recognition rate and lower recognition rate was shown in figure 8.

It is not difficult to find that the objects with higher recognition rate is more complexity than the objects with lower recognition rate.
4 Conclusion

In the present work, an optimized HTM method for multiple-object recognition and classification has been presented. The convolution method was used for capturing the basic features, while sparse representation was used in the HTM network. Moreover, the spatial module was replaced by the prediction space, thereby increasing the network speed for identifying the input patterns. Further, in order to successfully perform the image recognition task, the temporal module was changed into wakeup processing for child nodes effect upper level nodes. Finally, a three-level network was designed for locomotive object recognition, and the images of a locomotive bottom were heavily affected by waterlogging and dust. The recognition rate was approximately 91.4%.

References