The Comparison Of Different Graph Convolutional Neural Networks For Image Recognition

Sichao Fu, Xinghao Yang, Weifeng Liu
College of Information and Control Engineering
China University of Petroleum (East China), Qingdao
China
liuwf@upc.edu.cn

ABSTRACT

During the past decade, deep learning (DL) has been proven an effective way for image recognition. Various kinds of DL models such as, graph neural network (GNN), gated graph neural networks (GG-NNs) and graph convolutional neural networks (GCN) have been proposed. The latest GCN algorithm has achieved prominent performance for knowledge graph classification and molecular recognition. However, it only calculates the simple zero-one adjacency matrix, which is powerless to distinguish the self-node and other nodes. In this paper, we propose a novel negative one-zero-one graph to tackle this problem. Particularly, the negative one-zero-one graph employs zero to identify the node itself. To further explore the distance structure of inner-class samples, we develop the negative one-zero-one graph to a Euclidean distance version. Based on the ordinary graph, we also propose three kinds of hypergraph construction methods, such as hypergraph label method, hypergraph Euclidean distance method and the improvement method of hypergraph Euclidean distance. Additionally, we extend GCN to image recognition tasks in this paper. Extensive experiments are conducted on PASCAL VOC 2007 dataset to validate the proposed algorithms. The experiments results prove that the proposed methods is superior to the some methods for image recognition.

CCS CONCEPTS

• Computer vision/machine learning for multimedia application.

KEYWORDS

Graph convolutional neural network, graph theory, deep learning, hypergraph.

ACM Reference format:

Sichao Fu, Xinghao Yang and Weifeng Liu. 2018. The Comparison Of Different Graph Convolutional Neural Networks For Image Recognition.[1] In *The 10th International Conference on Internet Multimedia Computing and Service (ICIMCS'18)*, ACM, Nanjing, China, 6 pages. https://doi.org/10.1145/3240876.3240915

1 INTRODUCTION

During the past decade, we have witness the advent of big data era. With the exponentially increasing of image data, traditional image classification methods gradually show their limitations. As an alternation, deep learning (DL) technology [1-4] has been widely used in computer vision [5-8], speech recognition [9-12], video analysis [13-16], image classification [17-20], etc. image recognition task. However, DL still has a lot of room for development.

Recently, many deep learning algorithm have been proposed for image classification, such as graph neural network (GNN), gated graph neural networks (GG-NNs) and graph convolutional neural networks (GCN). Scarselli et al. [21] proposed a novel neural network model—GNN, it combine the recursive neural networks(RNN) and random walk models and save the advantages of RNN in graph domains. It was applied to deal with all kinds of graphs, such as directed graph, undirected graph. The realization mechanism of GNN has been used in Hopfield neural networks [22]. It also uses information diffusion mechanism, Graphs consist of many units, each units corresponding to one of the nodes in the graph. The nodes connected together according to the graph connectivity. These nodes constantly update their status until they reach a stable state. So there is a big restriction to this model and it is not applicable in many situations.

Li et al. [23] proposed a model named gated graph neural networks (GG-NNs) based on GNN. It use gated recurrent units and modern optimization techniques and expand GNN to outputs sequences. First, it use Recurrent Neural Networks (RNN) to learn the representation of the input graphs. Then, it is to learn features on graph and can get the output of the graph construction, which is not only individual classification. But it does not apply to large graph with a wide distribution of nodes.

N. Kipf et al. [24] proposed a novel approach named graph convolutional neural networks (GCN). It based on convolutional neural networks (CNN) and only add a graph theory. It introduces a layer-wise propagation rule for GCN and used each sample as a node in the graph. The construction of adjacency matrix is based on label category. If two different nodes belong to different class,

ICIMCS'18 August 17-19, 2018, Nanjing, China

ACM ISBN 978-1-4503-6520-8/18/08. . . $15.00
https://doi.org/10.1145/3240876.3240915

the distance of two nodes is set to one, else the distance of two nodes is zero. It reduce the computational complexity and change linearly as the number of edges. But the construction of adjacency matrix has one disadvantage that cannot distinguish the connection relationship between self-nodes.

To solve this problem, in this paper, we introduce a new approach, named different graph convolutional neural networks (DGCN). First, we introduce the negative one-zero-one graph construction method. Specifically, we add an additional idea that the distance between two self-nodes is zero. The main advantage of this method is that it can distinguish self-nodes with the nodes belong to different categories. Therefore, it reduces false positive rate. Second, to learn more about the connections between samples, we change the calculation method. Several sample's location that closest to a sample through Euclidean distance were calculated. If the location of one sample belong to this range, the distance is specific value, other cases the distance is zero. This method let the adjacency matrix of the graph become accurate and detailed.

In machine learning algorithms, we often assume that there is a certain relationship between two objects, and the relationship between objects is represented by an ordinary graph. However, in many real-world problems, the relationship between objects is not just a single relationship between two samples, but also a multiple or more complex relationship. If we simply represent the multiple relationship in a single relationship between two, it will inevitably lose a lot of important and useful information. Hypergraph are arbitrary sets of nodes, and therefore contain an arbitrary number of nodes. So the hypergraph is a generalization of a graph. Hypergraph not only effectively represent the complex relationship between objects, but also ensure the integrity and accuracy of the algorithm to some extent.

Based on the ordinary graph, we also propose three kinds of hypergraph construction methods. The first is hypergraph label method. We take the samples with the same class label as a hyperedge, such as, there are twenty categories in the PASCAL VOC 2007 [25] dataset, so there are twenty hyperedges. Then we can use the formula to calculate the distance between samples. The second is hypergraph Euclidean distance construction method. We take the several samples that closest to a sample through Euclidean distance were calculated as a hyperedge. Although the selected samples are closest to a certain sample, the labels of these samples may not be the same as the label of the central sample and thus increase the error rate. Therefore, the third method that we propose is the improvement method of hypergraph Euclidean distance construction method. We remove the sample that is not the same as the center sample class. In addition, in order to further prove the effectiveness of our proposed method, we validate the method on PASCAL VOC 2007 [25] dataset for image recognition, the experiments results demonstrate the advantages of our proposed method outperforms the some methods for image recognition.

The remainder of this article is organized as follows: In Section 2, we introduce our proposed algorithm. In Section 3, we introduce the experiments parameter and show the experimental results. Finally, we conclude this paper in section 4.

2 ALGORITHM

In this section, we first introduce the graph convolutional neural network (GCN) [24] that the others proposed. Then, we construct five different graph adjacency matrix to represent the relationships between samples. We combined the adjacency matrix of our proposed into graph convolutional neural network (GCN) for image classification, named different graph convolutional neural networks (DGCN). Below we introduce each section in turn.

2.1 Graph Convolutional Neural Network (GCN)

Defferrard et al. [26] proposed the definition of spectral convolutions on graphs, i.e. the convolution of a signal x with a filter g_θ, we now have:

$$g_\theta * x = \sum_{k=0}^{k} \theta_k T_k(\vec{L})x \tag{1}$$

Here, $\vec{L} = \frac{2}{\lambda_{\max}} L - I_N$.

N. Kipf et al. [24] proposed layer-wise convolution model, i.e the graph convolutions neural network model can be built through stacking multiple convolutional layers of the form of spectral , each layer accompany a point-wise non-linearity. It limit the layer-wise convolution operation to K = 1, get a linear function on the graph spectrum. Finally, it generalize this definition to a signal X with D-dimensional feature vector for every sample and F filters and denote as:

$$H^{(L+1)} = \sigma(\vec{D}^{-\frac{1}{2}} \vec{A} \vec{D}^{-\frac{1}{2}} H^{(L)} W^{(L)}) \tag{2}$$

Here, $\vec{A} = A + I_N$ is the adjacency matrix of the graph with added self-node, A is adjacency matrix that represent the relationship between different samples, $D_{ii} = \sum_j \vec{A}_{ij}$, I_N is identity matrix, and $W^{(L)}$ is one layer weight matrix. $\sigma(\ )$ is an activation function, such as RELU. $H^{(L)}$ is one layer sample feature matrix, $H^{(0)} = X$ .In addition, the construction of adjacency matrix is zero-one method in GCN and as follows:

Suppose $\{(x)_i\}_{i=1}^{n}$ denotes dataset and $\{(y)_i\}_{i=1}^{n}$ represents label class. Let M denote a set of samples, which belong to same class and N denote two samples belong to different class, i.e.

$$M = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ have the same labels}\} \tag{3}$$

$$N = \{(x_i, x_j) \mid x_i \text{ and } x_j \text{ have the different labels}\} \tag{4}$$

The adjacency matrix is the representation between samples. It has different construction methods. Currently, the zero-one adjacency matrix in graph convolutional neural networks (GCN) is the most popular one. If two nodes have same class, the distance is one. If two nodes have different class, the distance is zero. Based on the above relationship, we denote zero-one adjacency matrix A as:

$$A_{ij} = \begin{cases} 1 \quad \text{if} \left(x_i , x_j \right) \in M \\ 0 \quad \text{if} \left(x_i , x_j \right) \in N \end{cases} \tag{5}$$

2.2 Different Graph Convolutional Neural Network (DGCN)

But zero-one adjacency matrix have a shortcoming: it cannot distinguish the relationship between the node itself with other nodes (when sample do not have label, the relationships of the node itself and different class nodes both will be judged as zero and increase error rates.). So we propose the first graph construction methods: If two nodes belong to the same class but do not include nodes itself, the distance between two nodes is one. If two nodes belong to different class, the distance between two nodes is zero, else the distance is negative one. Suppose E denotes any two nodes have the same class but do not include the relationships between the nodes itself. Based on the above relationship, we denote negative one-zero-one adjacency matrix B1 as:

$$B1_{ij} = \begin{cases} 1 & if \left(x_i , x_j \right) \in E \\ 0 & if \left(x_i , x_j \right) \in N \\ -1 & else \end{cases} \tag{6}$$

Based on the relationship of the labels, both of the above methods use it to calculate the adjacency matrix. We further propose are based on the relationship of the feature between sample to calculate the distance. Therefore, we propose the second graph construction method that a method for calculating an adjacency matrix use the Euclidean distance. According to the Euclidean distance, the several closest samples of a sample are selected, so the distance of the several samples with this sample is specific value, the relationship of the other samples with this sample is zero. Suppose Q denotes the several closest samples, we denote Euclidean distance adjacency matrix B2 as:

$$B2_{ij} = \begin{cases} \sqrt{(x-y)^2} & \text{if} \left(x_i , x_j \right) \in Q \\ 0 & else \end{cases} \tag{7}$$

The above adjacency matrix belongs to the construction method of the ordinary graph. However, it simply represent the single relationship between two, so it will inevitably lose a lot of important and useful information. To solve this problem, we introduce the hypergraph construction methods. First, we introduce the formula for calculating hypergraphs adjacency matrix as follows: Suppose $V = \left\{ (v)_i \right\}_{i=1}^{n}$ denotes a set of nodes and $E = \left\{ (e)_i \right\}_{i=1}^{n}$ denotes a set of hyperedges.

$$C = HWH^T - D_V \tag{8}$$

Here, W is a diagonal matrix and the diagonal elements are the weights of each hyperedge. D_V is also a diagonal matrix and the diagonal elements are the degrees d(v) of each sample and as follows:

$$d(v) = \sum_{\{e \in E \mid v \in e\}} W(e) \tag{9}$$

H is a matrix expression of a hypergraph and as follows:

$$H = \begin{cases} 1 \quad \text{if } v \in e \\ 0 \quad else \end{cases} \tag{10}$$

The third graph construction method is hypergraph label. If some nodes belong to the same class, we take them as a hyperedge, i.e. the experimental dataset has several categories of labels and there are several hyperedges. The value of W is the total number of samples with the same label minus one. The adjacency matrix of the hypergraph label method B3 can be calculated by the above formula of 8.

The fourth graph construction method is hypergraph Euclidean distance. We take the several samples that closest to a sample through Euclidean distance were calculated as a hyperedge. The diagonal element value of W represents the sum of the nearest few samples Euclidean distances from the center node. The adjacency matrix of the hypergraph Euclidean distance method B4 can be calculated by the above formula of 8.

Although the selected samples are closest to a certain sample, the labels of these samples may not be the same as the label of the central sample and thus increase the error rate. So the fifth graph construction method is the improvement of hypergraph Euclidean distance. Thus, we remove the samples that the labels of these samples do not have the same as the label of the central sample. The diagonal element value of W represents the sum of the remaining nearest few samples Euclidean distances from the center node. The adjacency matrix of the improvement method of hypergraph Euclidean distance B5 can be calculated by the above formula of 8.

In this paper, we use a two layer different graph convolutional neural networks (DGCN) for semi supervised image classification. We can denote two layer DGCN as:

$$Z = \text{soft max}\left(\vec{B} RELU \left(\vec{B} X W^{(0)} \right) W^{(1)} \right) \tag{11}$$

Here, SOFTMAX function denote as:

$$S_j = \frac{e^{a_j}}{\sum_{k=1}^{T} e^{a_j}} \tag{12}$$

$W^{(0)}$ is weight matrix of the first layer , $W^{(1)}$ is weight matrix of the second layer, $\vec{B} = \vec{D}^{-\frac{1}{2}}\vec{B}\vec{D}^{-\frac{1}{2}}$, $\vec{B} = B+I_N$, B is adjacency matrix that represent the relationship between different samples . Our experiment environment is TensorFlow [27] based on GPU. The sample feature takes in the following input：

X: an N *D matrix is the description of sample feature, where N is the number of sample and D is the number of sample feature.

B: an N *N matrix is the description of sample adjacency distance.

In the process of training, the weights up-to-date use gradient descent. The cross entropy error L that we used is:

$$L = -\sum_{j=1}^{T} y_i \log Z_j \tag{13}$$

Here, Z is the output after activation function.

3 EXPERIMENT

In this section, we conduct experiments on PASCAL VOC 2007 dataset [25] to prove the validity of our proposed algorithm for image classification. First, we introduce the PASCAL VOC 2007 database [25]. Then, we describe our experiments setting. Finally, we show our experimental results.

3.1 PASCAL VOC 2007 Dataset

The PASCAL VOC 2007 dataset [25] has a total of 9963 images and all images are 500*375 or 375*500 pixels. It divided into 20 different categories in Figs. 1 (i.e. aero, bicycle, bird, boat, bottle, bus, car, cat, chair, cow, dinningtable, dog, horse, mbike, person, plant, sheep, sofa, train, tv). It include indoor scenes, outdoor scenes, close-ups, landscapes and strange viewpoints. The dataset is extremely challenging because all the images that the size, viewing angle, illumination and the appearances of objects are vary significantly. In [30], it proposed 15 different image representations methods, such as SIFT, Hue, Rgb, Hsv, Lab and DenseSIFTV3H1. We only used a single descriptor, the DenseSIFTV3H1, in the experiment, i.e. it is to compute a 3 × 1 horizontal decomposition of the image through encoding some spatial information.

Figure 1: PASCAL VOC 2007 dataset.

3.2 Experiments Setting

In this section, we use the DGCN model that our proposed in section II and use it for image classification through semi-supervised learning. We compare the GCN model (zero-one adjacency matrix) with the DGCN model (negative one-zero-one adjacency matrix, Euclidean distance adjacency matrix, hypergraph label adjacency matrix, hypergraph Euclidean distance adjacency matrix and the improvement adjacency matrix of hypergraph Euclidean distance). Due to the large number of the adjacency matrix data, our hardware configuration is too low to handle and memory overflow is very easy. We can only choose a sample with fewer data sets. In this paper, we do experiment on the PASCAL VOC 2007 [25] dataset and it is divided into three parts: training set, validation set, and test set. We split the 9963 images into a training set of 5979 images, a validation set of 1992 images and a test set of 1992 images. In addition, the label rate is set to 10%, 15%, 20%, 25% and 30% in training set. We optimize the weight through the Adam [28] with the learning rate of 0.00001. If the loss value of the validation set does not drop for 10 consecutive times, our model stops training immediately. We use the weight initialization method described in [29]. The weight for L2 loss on embedding matrix is 5e-4. Other parameters include the number of units in hidden units, the dropout rate and the maximum value of the training iteration during the training procedure. We use the following sets of parameters for Euclidean distance method, hypergraph Euclidean distance method and the improvement method of hypergraph Euclidean distance: 0.2 (dropout rate), 200 (maximum value of the training iteration) and 64 (number of hidden units); and for zero-one method, negative one-zero-one method and hypergraph label method: 0.5 (dropout rate), 300 (maximum value of the training iteration) and 16 (number of hidden units).

We measure the algorithm performance using the average precision (AP) criterion for each class and the mean AP (MAP) criterion over all classes. The definition of PASCAL VOC 2007 challenge [25] evaluation method is as follows:

$$AP = \frac{1}{11}\sum_{r} P\left(r\right) \tag{14}$$

Here, P(r) is the maximum precision over all recalls larger than r ∈ {0,0.1,0.2,…,1.0}.

3.3 Experiments Results

In this section, we compare the zero-one adjacency matrix with the method of negative one-zero-one, Euclidean distance, hypergraph label, hypergraph Euclidean distance and the improvement of hypergraph Euclidean distance in turn. The results of the experiment are summarized in the Figs. 2. From the experimental results we can know, the best in the six methods is hypergraph label in most cases. In addition, hypergraph label also is the best in the label methods in most cases. Of all Euclidean distance methods, Euclidean distance is the best. In the Figs. 3, Figs. 4, Figs. 5, Figs. 6 , Figs. 7 and Figs.8, we list the AP performance of our proposed method in each type of sample. Due to space reason, we can only randomly choose six categories.

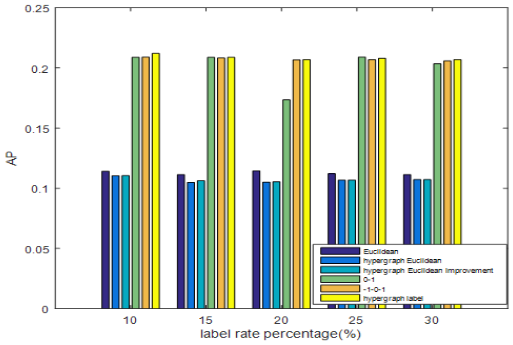

Figure 2: Performance in MAP of the six compared methods

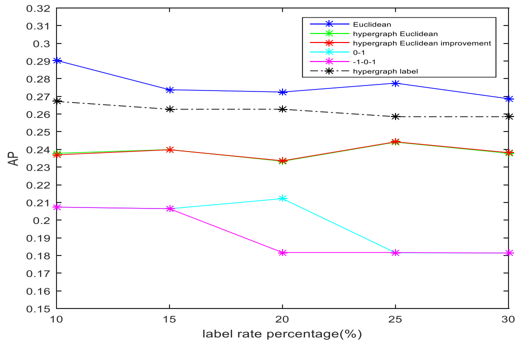

Figure 5: Performance in AP of the six compared methods in car category

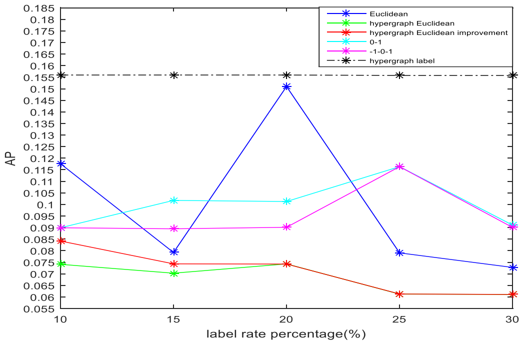

Figure 3: Performance in AP of the six compared methods in bicycle category

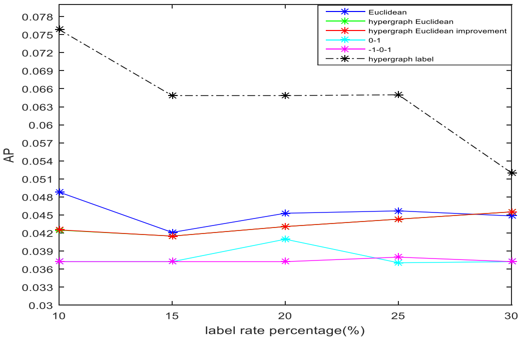

Figure 6: Performance in AP of the six compared methods in cow category

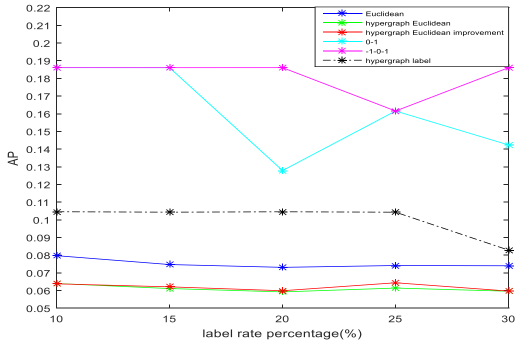

Figure 4: Performance in AP of the six compared methods in boat category

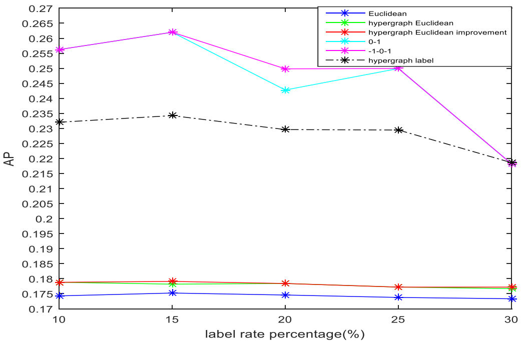

Figure 7: Performance in AP of the six compared methods in dog category

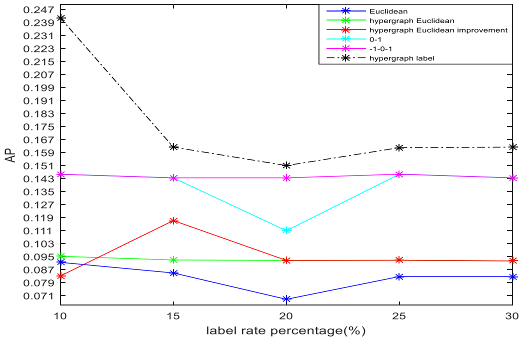

Figure 8: Performance in AP of the six compared methods in train category

4 CONCLUSIONS

In these years, the study of graph theory has became a hot research topic in deep learning area. A proper distance calculating methods is vital important for delivering a high recognition accuracy. Until now, the graph theory with deep learning have been combined and many algorithms have been proposed. The most popular one is graph convolutional neural networks (GCN), which is a variant of convolutional neural networks (CNN). In GCN, the construction of adjacency matrix is zero-one, which cannot distinguish the relationship of self-nodes. To solve this problem, in this paper, we introduce two methods for building adjacency matrix. The methods that our proposed are the adjacency matrix of negative one-zero-one and Euclidean distance. In addition, we also propose the construction method of hypergraph adjacency matrix, such as hypergraph label method, hypergraph Euclidean distance method and the improvement method of hypergraph Euclidean distance. The experiments results validate that the proposed methods outperforms the other methods for image recognition.

ACKNOWLEDGMENTS

This work is partially supported by the National Natural Science Foundation of China (Grant No. 61671480), the Foundation of Shandong province (Grant No. ZR2018MF017) and the Fundamental Research Funds for the Central Universities, China University of Petroleum (East China) (Grant No. 18CX07011A).

REFERENCES

[1] F Clément, C Camille, N Laurent and L Yann (2013). Learning hierarchical features for scene labeling. IEEE transactions on pattern analysis and machine intelligence, 35(8), 1915-1929.

[2] B Yoshua, D Olivier and L.R Nicolas (2006). The curse of highly variable functions for local kernel machines. Advances in neural information processing systems，107-114.

[3] V Pascal, L Hugo, B Yoshua and M Pierre-Antoine (2008). Extracting and composing robust features with denoising autoencoders. Proceedings of the 25th international conference on Machine learning, 1096-1103.

[4] M Tomas (2010). Recurrent neural network based language model. Eleventh Annual Conference of the International Speech Communication Association.

[5] D Haluk and H Elliott (1987). Modeling and segmentation of noisy and textured images using Gibbs random fields. IEEE Transactions on pattern analysis and machine intelligence, 1987 (1), 39-55.

[6] L Stan Z (1992). Toward 3D vision from range images, an optimization framework and parallel networks. CVGIP, 55(3), 231-260.

[7] M Jose, S Mitter, and T Poggio (1987). Probabilistic solution of ill-posed problems in computational vision. Journal of the american statistical association, 82(397), 76-89.

[8] A Jörg (2010). Multi stereo camera data fusion for fingertip detection in gesture recognition systems. Soft Computing and Pattern Recognition (SoCPaR), 35-40.

[9] Y GuoQing and B Dietmar (2010). Daily activity learning from motion detector data for ambient assisted living. Human System Interactions (HSI), 89-94.

[10] K Gerhard, L Urs, R Jurg and J Pierre (1996). A comparison of demodulation techniques for code time division multiple access. Global Telecommunications Conference, 525-529.

[11] W Ehsan Ullah and K Tei (2013). Fault detection in wireless sensor networks, a machine learning approach. Computational Science and Engineering (CSE), 758-765.

[12] L Kai-Fu and H Hsiao-Wuen (1989). Speaker-independent phone recognition using hidden Markov models. IEEE Transactions on Acoustics, Speech, and Signal Processing, 37(11), 1641-1648.

[13] M Dan (1995). Ethnography and ethnographic film, From Flaherty to Asch and after. American Anthropologist, 97(2), 339-347.

[14] S Albert E (1964). The significance of posture in communication systems. Psychiatry, 27(4), 316-331.

[15] B Gregory (1958). Language and psychotherapy, Frieda Fromm-Reichmann's last project. Psychiatry, 21(1), 96.

[16] L Fabien and C Schnörr (2009). Spectral clustering of linear subspaces for motion segmentation. IEEE Int. Conf. on Computer Vision (ICCV'09).

[17] D.P Du, X.Y Xu, T.W Ren and G.S Wu. Depth images could tell us more, enhancing depth discriminability for rgb-d scene recognition.

[18] Z Xiantong and L Shao (2014). A performance evaluation on action recognition with local features. Pattern Recognition (ICPR), 4495-4500.

[19] J.C Yang, K Yu, Y.H Gong and T Huang (2009). Linear spatial pyramid matching using sparse coding for image classification. Computer Vision and Pattern Recognition, 1794-1801.

[20] Y Kai, T Zhang and Y.H Gong (2009). Nonlinear learning using local coordinate coding. Advances in neural information processing systems, 2223-2231.

[21] S Franco (2009). The graph neural network model. IEEE Transactions on Neural Networks, 20(1), 61-80.

[22] H John J (1982). Neural networks and physical systems with emergent collective computational abilities. Proceedings of the national academy of sciences, 79(8), 2554-2558.

[23] L Yujia and Z Richard (2015). Gated graph sequence neural networks. arXiv preprint arXiv:1511.05493.

[24] K Thomas N and M Welling (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

[25] E Mark (2010). The pascal visual object classes (voc) challenge. International journal of computer vision, 88(2), 303-338.

[26] D Michaël, X Bresson and P Vandergheyns (2016). Convolutional neural networks on graphs with fast localized spectral filtering. Advances in Neural Information Processing Systems, 3844-3852.

[27] A Martn (2015). TensorFlow: large-scale machine learning on heterogeneous systems. Software available from tensorflow. URL https://www. tensorflow. org.

[28] K D and J.B Adam (2015). A method for stochastic optimization. International Conference on Learning Representations (ICLR).

[29] G Xavier and Y Bengio (2010). Understanding the difficulty of training deep feedforward neural networks. Proceedings of the thirteenth international conference on artificial intelligence and statistics, 249-256.

[30] G Matthieu, J Verbeek and C Schmid (2010). Multimodal semi-supervised learning for image classification. Computer Vision and Pattern Recognition (CVPR), 902-909.