Advances in Modelling and Analysis B
Vol. 61, No. 3, September, 2018, pp. 158-162
Journal homepage: http://iieta.org/Journals/AMA/AMA_B

Deep feedforward neural network learning using Local Binary Patterns histograms for outdoor object categorization

Heni Bouhamed[1*], Yassine Ruichek[2]

[1] Advanced Technologies for Image and Signal Processing unit, Technopole of Sfax, 3018, Tunisia
[2] Le2i FRE2005, CNRS, Arts et Métiers, University Bourgogne Franche-Comté, UTBM, Belfort F 90010, France

Corresponding Author Email: heni.bouhamed@fsegs.usf.tn

https://doi.org/10.18280/ama_b.610309

Received: 4 July 2018
Accepted: 25 August 2018

Keywords:
deep learning, deep feedforward neural network, local binary pattern histogram, classification

ABSTRACT

Advanced driver assistance systems and outdoor video surveillance very often need to classify the detected objects/obstacles. In this context several works have presented and have tested some graph-based methods. Motivated by the prominence of deep neural networks, which surpass the performance of the previous dominating paradigm, we are going to apply him in the classification of images by using the local binary pattern (LBP) histograms, to our knowledge, our work is the only one to propose this conduct. We go to see that the results are very promising besides the fact that the construction of such a model is possible also in a massive data context.

1. INTRODUCTION

Advanced driver assistance systems and outdoor video surveillance very often need to classify the detected objects/obstacles. The considered classes (labels) are the various answers according to the degree of the situation importance. The information of classification can be integrated into the global architecture of the navigation assistance for example in obstacle avoidance, a module/object following etc. In the systems of driver assistance for the commercial cars, the information of classification can be used to trigger alarms or the corresponding action [5]. There were two main categories of approaches developed based on the visual data, the first one uses a trained detector for a specific class [9]. The second category of approach makes a detection phase before considering the class of the detected object. The first category of approach can be applied when the application concentrates on a single class, nevertheless, it becomes difficult to apply when there are several classes to be simultaneously considered. The second category of approach can be deployed according to the number of classes which the system will have to recognize.

In this context several works have presented and have tested some graph-based methods as the K Nearest Neighbor method (KNN), the Locally Linear Embedding (LLE) and the Two phase weighted regularized least square graph construction (TPWRLS) etc. [5-8, 13, 20]. Motivated by the prominence of deep neural networks, which surpass the performance of the previous dominant paradigm [4, 11, 17, 19], we suggest applying it in the classification of images by using the local binary pattern (LBP) histogram [12]. Several works [5-8, 13, 20] has shown the effectiveness of using the notion of lbp patterns frequencies (histogram) (Figure 2) for images classification. This behavior can reduce the number of features to 59 when using only lbp uniform patterns (regardless of the size of the images).

The deep learning proposes several architectures which can be used according to the context. The most general of them is called "Deep feedforward neural network (DFFNN)" [10], its name badge the obligation that neurons have only a forward distribution (Figure 1). The performances of this architecture already exceed the machine learning classic dominant paradigms (gaussian mixture model, naïve Bayesian classifier, decision trees, knn etc.) in several applications [4, 11, 17, 19], in this sense, we have chosen to start with this non-specialized architecture and we are leaving open the option to the use of other architectures in case of unsatisfactory results.

The remainder of this paper was organized as follows. Section 2 was devoted to the DFFNN and their architecture. However, the LBP was introduced in Section 3 before dealing with methodologies and performance evaluation in Section 4. Our conclusion and perspectives were drawn in the last section.

2. DEEP FEEDFORWARD DEEP NEURAL NETWORK (DFFNN)

The deep learning is a set of machine learning methods allowing to model data with a high level of abstraction. It is based on articulate architectures of various transformations in the no linear space [2]. Is considered a part (or a complement) to the Big Data domain. Current interest for the deep learning is not only for his conceptual advances but also for the technological advancess, indeed, all the actually available serious solutions (in terms of models learning) are capable to exploit the immense reservoir of power computing established through actual modern computers, as well by requesting the main processor (CPU) that the graphic dedicated processors (GPU). A model Big Data is capable of adapting itself when there is an enormous volume of data to be handled or when there is an enormous sequential treatment numbers exceeding the most powerful servers capacities [22].

Recent findings in the field of image and speech recognition have shown that significant accuracy improvements over classical schemes (as gaussian mixture model, decision trees, KNN etc.) can be achieved through the use of DFFNN [4, 11, 17, 19]. DFFNN can be used as classifiers that directly estimate class posterior scores. Among the most important advantages of DFFNN is their multilevel distributed representation of the model's input data [11].

This fact makes the DFFNN an exponentially more compact model than GMMs. Further, DFFNN do not impose assumptions on the input data distribution [17] and have proven successful in exploiting large amounts of data, achieving more robust models without lapsing into overtraining. All of these factors motivate the use of DFFNN for outdoor object categorization.

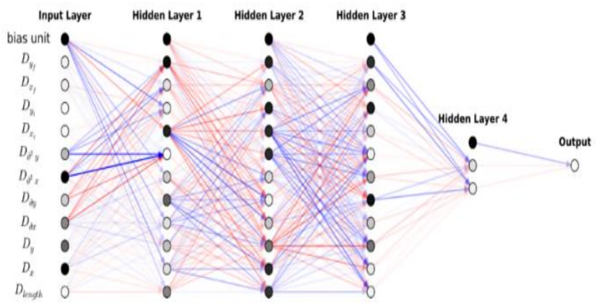

Figure 1. Example of deep feedforward neural network

The DFFNN system used in this work is a fully-connected feed-forward neural network with rectified linear units (ReLU) [21].

Thus, an input at level j, x_j, is mapped to its corresponding activation y_j (input of the layer above) as:

$$y_j = ReLU(x_j) = \max(0, x_j) \tag{1}$$

$$x_j = b_j + \sum_i w_{ij} y_i \tag{2}$$

where i is an index over the units of the layer below and b_j is the bias of the unit j.

The output layer is then configured as a softmax, where hidden units map input y_j to a class probability p_j in the form:

$$p_j = \frac{exp(y_j)}{\sum_l exp(y_l)} \tag{3}$$

where l is an index over all of the target classes.

As a cost function for backpropagating gradients in the training stage, we use the cross-entropy function defined as:

$$C = -\sum_j t_j \log p_j \tag{4}$$

where t_j represents the target probability of the class j for the current evaluated example, taking a value of either 1 (true class) or 0 (false class) [16].

3. LOCAL BINARY PATTERNS

The original LBP operator labels the pixels of an image with decimal numbers, which are called LBPs or LBP codes that encode the local structure around each pixel [12]. It proceeds thus, as illustrated in Figure 2a: Each pixel is compared with its eight neighbors in a neighborhood by subtracting the central pixel value; the resulting strictly negative values are encoded with 0, and the others with 1. For each given pixel, a binary number is obtained by concatenating all these binary values in a clockwise direction, which starts from the one of its top-left neighbors. The corresponding decimal value of the generated binary number is then used for labeling the given pixel. The histogram of LBP labels (the frequency of occurrence of each code) calculated over a region or an image can be used as a texture descriptor [5].

The neighbors of the central pixel can be simply the direct neighbors (radius=1) or in other cases the 2 units apart pixels neighbors (radius = 2) or even the 3 units apart pixels neighbors (radius = 3). The neighbor numbers can vary also, 8 at the most if the radius is equal to 1 and more from the radius 2. We have chosen during this work to opt for a number of neighbors equal to 8 with the 3 first possible radius (r = 1, r = 2 and r = 3). We can, in future research, test more combinations with a more important number of neighborhood.

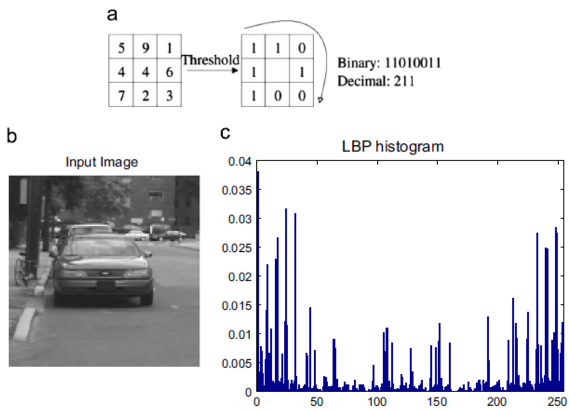

Figure 2. LBP from input to histogram

4. METHODOLOGIES AND PERFORMANCE EVALUATION

Usually, when using different deep learning architectures in image recognition, the input often used is the different pixels forming images. The number of elements in input may be quite important, for example, an image with a 1000/1000 size will be considered as an input with 1000000 entries, which may be a problem during learning process especially when you have a massive data. Several works [5-8, 13, 20] has shown the effectiveness of using the notion of lbp patterns frequencies (histogram) (Figure 2) to build a graph-based models for classification. This behavior can reduce the number of entries to 59 when using only lbp uniform patterns (regardless of the size of the images). The Figure 3 shows the different cases where the pattern can be uniform (a one single change of the binary digits). We propose in our work to use this pipe but with the DFFNN instead of simple graph-based methods used in the specialized literature.

We are going to evaluate DFFNN architecture (Codes are developed in python 3.6 with Tensorflow Backend) for the objects categorization by means of the cross-validation scheme that is commonly used in the domain of pattern recognition. To this end, the whole data set is split into two parts: a part with known labels (usually called training set) and a part with unknown labels (called test set). Note that the ground-truth labels of the latter set are used in order to estimate the rate of correct classification. The accuracy of label inference is evaluated by comparing the estimated labels with the ground-truth ones. This process is repeated ten times in order to get statistical stability in the evaluation of the given formalism. In each trial, the set is randomly split into a labeled part and an unlabeled part. The accuracy is given as an average over the ten random splits. Objects can be captured by either a surveillance camera or an onboard camera. We assume that the detection of the image regions containing the object is carried out by an algorithm as those described in [1, 14-15] for the case of surveillance cameras or by the algorithms of detection and tracking as those described in [3, 18] for the case of an onboard camera.

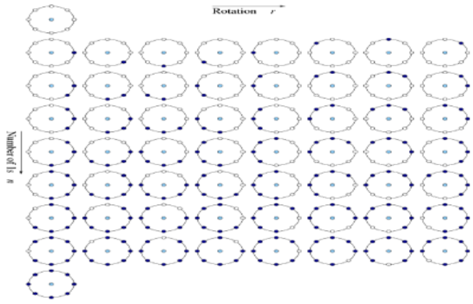

Figure 3. Uniform LBP pattern

In the following part, we are going to present a quantitative evaluation comparing the DFFNN and some graph-based methods in the task of objects categorization [5-8, 13, 20]. This conduct is applied, firstly, to outdoor object categorization using a first public outdoor image dataset, and secondly, to object categorization using a second public dataset. We performed two groups of experiments. In the first group, we used images presenting three classes (Pedestrian, cars/vans, and motorbikes) (see Figure 4). The car and moto images were obtained from PASCAL VOC2011 Examples Images (http://pascallin.ecs.soton.ac.uk/challenges/VOC/voc2012/examples/index). The pedestrian images are obtained from CVC-01 (http://www.cvc.uab.es/adas) Classification Dataset (images, of this group of experiments, have variable sizes). We gathered 450 images (150 images per class) Their LBP descriptors were computed using the uniform patterns (r = 1, 2 and 3 with a neighborhood equal to 8 in each case). Table 1, 2, 3 illustrates precision, recall and f1-measure, for each label, obtained by inferring DFFNNs models according to the different types de neighborhoods (see Figure 6 for more details concerning precision, recall and f1-measure formulas).

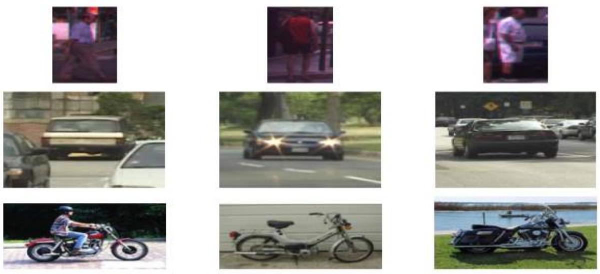

Figure 4. Images presenting three classes (Pedestrian, cars/vans, and motorbikes)

Table 4 illustrates the accuracy obtained with DFFNN and some graph-based methods (knn, LLE, TPWRLS) applied on the same dataset. These are average results that correspond to ten runs of the recognition algorithm with random partitions for labeled and unlabeled samples. To note that the correct classifications rate of this some graph-based methods (on the same databases) were taken from [5] tests.

Table 1. Precision, recall and f1-measure, for radius = 1, obtained by inferring DFFNN model

r = 1	Precision	recall	f1-score
Cars/Vans	100	100	100
Motos	100	100	100
Pedestrians	100	100	100
Average	100	100	100

Table 2. Precision, recall and f1-measure, for radius = 2, obtained by inferring DFFNN model

r = 2	precision	recall	f1-score
Cars/Vans	98,3	98,7	98,4
Motos	98,6	98,5	98,4
Pedestrians	100	99,7	99,8
Average	98,96	98,96	98,86

Table 3. Precision, recall and f1-measure, for radius = 3, obtained by inferring DFFNN model

r =3	precision	recall	f1-score
Cars/Vans	95,2	99,3	97,2
Motos	99,3	95,2	97,1
Pedestrians	100	100	100
Average	98,16	98,16	98,1

Table 4. Average accuracy (first database)

Data bases 1	R = 1	R = 2	R = 3
KNN	90,9	95,9	95,8
LLE	93,8	96,5	97,3
TPWRLS	95,7	97,9	97,5
DFFNN	100	98,966	98,166

We can observe that the accuracy is much better than those of the graph-based methods already used in this context, indeed, the results are even perfect for r=1. The results of precision, recall and of f1-measure are very close to 100% for r=1 and r=2, nevertheless, their precision begins to slightly yield from r=3. We can conclude that there is really a clear results improvement by using the DFFNN. We still have to validate this improvement with the second database.

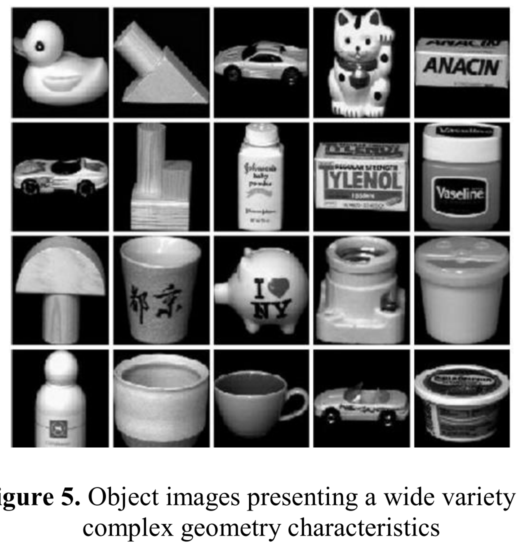

Figure 5. Object images presenting a wide variety of complex geometry characteristics

For the second group of experiments, the COIL-20 (http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php) database (Columbia Object Image Library) consists of 1440 images of 20 objects (images, of this group of experiment, have the same size). Each object has 72 images (each object has underwent 72 rotations). The object presents a wide variety of complex geometry characteristics. Some examples are shown in Figure 5. Their LBP descriptors have computed using the uniform patterns (r=1, 2 et 3 with a neighborhood equal to 8 in each case). Table 5,6,7 illustrates precision, recall and f1-measure, for each label, obtained by inferring DFFNNs models according to the different types de neighborhoods.

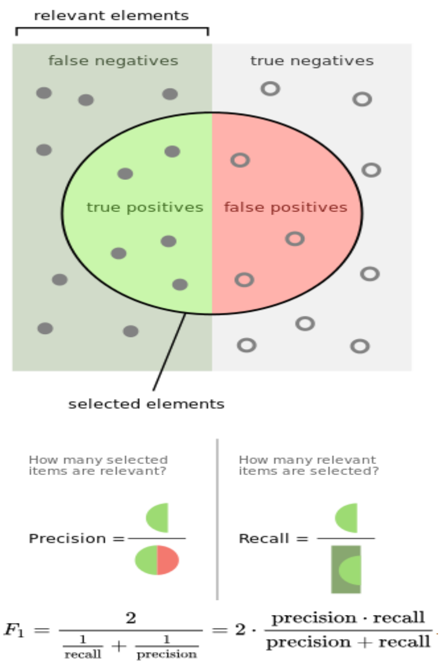

Figure 6. Precision, recall and f1-measure formulas.

Table 5. Precision, recall and f1-measure, for radius = 1, obtained by inferring DFFNN model

r = 1	Precision	recall	f1-score
Object 1	100	100	100
Object 2	100	98,8	99,3
Object 3	100	100	100
Object 4	100	100	100
Object 5	98,7	99,3	98,9
Object 6	100	98,8	99,3
Object 7	99,5	100	99,7
Object 8	100	100	100
Object 9	99,3	100	99,6
Object 10	100	100	100
Object 11	100	99,4	99,7
Object 12	100	100	100
Object 13	100	100	100
Object 14	100	100	100
Object 15	100	100	100
Object 16	100	100	100
Object 17	100	100	100
Object 18	100	100	100
Object 19	98,6	100	99,2
Object 20	100	100	100
Average	99,805	99,815	99,8

Table 6. Precision, recall and f1-measure, for radius = 1, obtained by inferring DFFNN model

r = 2	Precision	recall	f1-score
Object 1	100	100	100
Object 2	100	100	100
Object 3	100	100	100
Object 4	100	100	100
Object 5	100	100	100
Object 6	99,3	100	99,6
Object 7	100	100	100
Object 8	100	100	100
Object 9	100	100	100
Object 10	100	100	100
Object 11	100	100	100
Object 12	100	100	100
Object 13	100	100	100
Object 14	100	100	100
Object 15	100	100	100
Object 16	100	100	100
Object 17	100	100	100
Object 18	100	100	100
Object 19	100	99,3	99,6
Object 20	100	100	100
Average	99,965	99,965	99,96

Table 7. Precision, recall and f1-measure, for radius = 1, obtained by inferring DFFNN model.

r = 3	Precision	recall	f1-score
Object 1	100	100	100
Object 2	100	100	100
Object 3	95,6	98,6	96,5
Object 4	99,4	100	99,7
Object 5	100	100	100
Object 6	97,8	91,6	93,6
Object 7	99,2	100	99,6
Object 8	100	100	100
Object 9	100	100	100
Object 10	100	100	100
Object 11	100	99,4	99,7
Object 12	100	100	100
Object 13	100	100	100
Object 14	100	100	100
Object 15	100	100	100
Object 16	100	100	100
Object 17	100	100	100
Object 18	100	100	100
Object 19	97,9	98,8	98,3
Object 20	100	100	100
Average	99,495	99,42	99,37

Table 8 illustrates the accuracy obtained with DFFNN and some graph-based methods (knn, LLE, TPWRLS) applied on the same dataset. These are average results that correspond to ten runs of the recognition algorithm with random partitions for labeled and unlabeled samples. To note that the correct classifications rate of this some graph-based methods (on the same databases) were taken from [5] tests. [5] considered according to their experiments that the results found with r=2 and the neighborhood of 8 are best and that's why the results for r=1 and r=3 were not published.

We can observe that the accuracy is much better than those of the graph-based methods already used in this context, indeed, the results are almost perfect for r = 1, r = 2 and even for r=3. The results of precision, recall and of f1-measure are very close to 100% for r=1 and r=2, nevertheless, their precision begins to slightly yield from r=3. We can conclude that there is a really clear improvement of the results by using the DFFNN, this improvement is more sensitive on this database with regard to the first one.

From our two-step experiments, we have been able to show the superiority of the DFFNNs compared to the graph-based methods used in this context. Among the strong points, too, of our conduct is that the construction of such a model is very feasible also in a massive data context.

Table 8. Average accuracy (second database)

Second database	R = 1	R = 2	R = 3
KNN	-	90,58	-
LLE	-	95	-
TPWRLS	-	97,33	-
DFFNN	99,805	99,965	99,495

5. CONCLUSIONS

We have evaluated the DFFNN for the objects categorization with the cross-validation scheme that is commonly used in the domain of pattern recognition. Objects can be captured by either a surveillance camera or an onboard camera. In this work, we have presented a quantitative evaluation using the DFFNN and some graph-based methods schemes, applied, firstly to outdoor object categorization using a first public outdoor image dataset, and secondly, to object categorization using a second public dataset. From our two-step experiments, we have been able to show the superiority of the DFFNNs compared to the graph-based methods used in this context. Among the strong points, too, of our conduct is that the construction of such a model is very feasible also in a massive data context. It is in our perspective for future research to test this architecture with other LBP neighborhood types on a real data captured directly from a surveillance camera or an onboard camera.

REFERENCES

[1] Albusac J, Castro-Schez JJ, López-López LM, Vallejo D, Jimenez-Linares L. (2009). A supervised learning approach to automate the acquisition of knowledge in surveillance systems. Signal Processing 89(12): 2400-2414.

[2] Bengio Y. (2009). Learning deep architectures for AI. Foundations and trends® in Machine Learning 2(1): 1-127.

[3] Cheng B, Yang J, Yan S, Fu Y, Huang TS. (2010). Learning With ℓ^{1}-Graph for Image Analysis. IEEE Transactions on Image Processing 19(4): 858-866.

[4] Cireşan DC, Meier U, Gambardella LM, Schmidhuber J. (2010). Deep, big, simple neural nets for handwritten digit recognition. Neural Computation 22(12): 3207-3220.

[5] Dornaika F, Bosaghzadeh A, Salmane H, Ruichek Y. (2014). A graph construction method using LBP self-representativeness for outdoor object categorization. Engineering Applications of Artificial Intelligence 36: 294-302.

[6] Dornaika F, Bosaghzadeh A, Salmane H, Ruichek Y. (2014). Graph-based semi-supervised learning with Local Binary Patterns for holistic object categorization. Expert Systems with Applications 41(17): 7744-7753.

[7] Dornaika F, Bosaghzadeh A. (2015). Adaptive graph construction using data self-representativeness for pattern classification. Information Sciences 325: 118-139.

[8] Dornaika F, Moujahid A, El Merabet Y, Ruichek Y. (2016). Building detection from orthophotos using a machine learning approach: An empirical study on image segmentation and descriptors. Expert Systems with Applications 58: 130-142.

[9] Geronimo D, Lopez AM, Sappa AD, Graf T. (2010). Survey of pedestrian detection for advanced driver assistance systems. IEEE transactions on pattern analysis and machine intelligence 32(7): 1239-1258.

[10] Goodfellow I, Bengio Y, Courville A, Bengio Y. (2016). Deep learning. Cambridge: MIT press 1.

[11] Hinton G, Deng L, Yu D, Dahl GE, Mohamed AR, Jaitly N, Kingsbury B. (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal processing magazine 29(6): 82-97.

[12] Huang D, Shan C, Ardabilian M, Wang Y, Chen L. (2011). Local binary patterns and its application to facial image analysis: A survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 41(6): 765-781.

[13] Jebara T, Wang J, Chang SF. (2009). Graph construction and b-matching for semi-supervised learning. In Proceedings of the 26th annual international conference on machine learning 441-448. ACM.

[14] Kim K, Chalidabhongse TH, Harwood D, Davis L. (2005). Real-time foreground–background segmentation using codebook model. Real-time imaging 11(3): 172-185.

[15] Kim W, Kim C. (2012). Background subtraction for dynamic texture scenes using fuzzy color histograms. IEEE Signal processing letters 19(3): 127-130.

[16] Lopez-Moreno I, Gonzalez-Dominguez J, Martinez D, Plchot O, Gonzalez-Rodriguez J, Moreno PJ. (2016). On the use of deep feedforward neural networks for automatic language identification. Computer Speech & Language 40: 46-59.

[17] Mohamed AR, Dahl GE, Hinton G. (2012). Acoustic modeling using deep belief networks. IEEE Trans. Audio, Speech & Language Processing 20(1): 14-22.

[18] Pan P, Schonfeld D. (2011). Visual tracking using high-order particle filtering. IEEE Signal Processing Letters 18(1): 51-54.

[19] Yu D, Deng L. (2011). Deep learning and its applications to signal and information processing [exploratory dsp]. IEEE Signal Processing Magazine 28(1): 145-154.

[20] Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y. (2009). Robust face recognition via sparse representation. IEEE transactions on pattern analysis and machine intelligence 31(2): 210-227.

[21] Zeiler MD, Ranzato M, Monga R, Mao M, Yang K, Le QV, Hinton GE. (2013). On rectified linear units for speech processing. In Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on 3517-3521. IEEE.

[22] Zikopoulos P, Eaton C. (2011). Understanding big data: Analytics for enterprise class hadoop and streaming data. McGraw-Hill Osborne Media.