Sparse Wavelet Auto-Encoders for Image classification

Salima Hassairi, Ridha Ejbali and Mourad Zaied
REGIM-Lab: REsearch Groups in Intelligent Machines
University of Sfax, National Engineering School of Sfax (ENIS), BP 1173, Sfax, 3038, Tunisia
salima.hassairi.tn@ieee.org, ridha_ejbali@ieee.org, mourad.zaied@ieee.org

Abstract**—The goal of the Deep learning methods is learning feature hierarchies with features from higher levels to lower level features of the hierarchy. The major contribution of this paper is to show how to extract features and train an image classification system on large-scale datasets. This method is an improvement of our recent work. The training is carried out by the combination of the most used methods for image classification: Deep Learning and the Wavelet Network. Some algorithms of DL like the sparse coding and the stacked autoencoders are used in our approach. For the WN, the Fast Wavelet Transform and the Best Contribution Algorithm are utilized. The ImageNet dataset used in the test phase in which we used many criteria such as the number of the hidden layers and the number of images that we specified for the training shows great efficiency of our model for image classification compared to another approach.**

Keywords—Deep Learning; Wavelet Networks; Feature extraction; Image classification;

I. INTRODUCTION

All classification algorithms are centered on the hypothesis that the image in question depicts one or more features. Each of these features belongs to one of several distinct and exclusive classes [1, 2]. So, the feature extraction is an old problem in the field of image classification. However, it has been the most fundamental and decisive problem in the field. So far, there have been several kinds of image feature extractions for image classification such as the Deep Learning (DL) and the Wavelet Network (WN) algorithms.

Recently, several extraction methods of features are employed in the DL by some researchers. Among these methods, we cite the stacked autoencoders and the sparse coding algorithm [3-5].

The stacked autoencoders [6, 7] are a neural network consisting of multiple layers of autoencoders (AEs). The outputs of each layer in the AEs are connected to the inputs of the successive AE. A good way to obtain good parameters for a stacked autoencoder is to use the greedy layer-wise training. This method trains the parameters of each layer individually while freezing parameters for the rest of the model. To produce better results, after this phase of training is complete, fine-tuning using backpropagation can be used to improve the results by tuning the parameters of all the layers that are changed at the same time.

The aim of sparse coding [8] is to find a set of basis vectors ϕ_i. As such, we can represent an input vector X as a linear combination of these basis vectors. Sparse coding is the representation of items by the strong activation of a relatively small set of neurons. For each stimulus, this is a different subset of all available neurons [9,10].

All these approaches utilize neural networks. Yet, our objective is to integrate wavelet networks in these methods. So, the Wavelet Neural Networks [11,12] are the combination of two theories: the wavelets and the neural networks. A wavelet neural network uses one hidden layer and consists of a feed-forward neural network whose taking of one or more inputs and the output layer consists of one or more linear combiners or summers. The hidden layer consists of neurons whose activation functions are drawn from a wavelet basis. A wavelet network is defined by pondering a set of wavelets dilated and translated from one mother wavelet with weight values to approximate a given signal f [13, 14].

In this paper, we add a new technique in the training step of our algorithm proposed in previous works to supervised image classification with WN and DL methods. The part that remained of this paper contains five sections. Section 2 includes an overview of the recent work. The proposed method is presented in Section 3. The experiments and the tests' results are mentioned in Section 4. As a final point, section 5 concludes this paper.

II. RECENT WORK

In the Previous work, one of our objective was the combination of two theories: Deep Learning and Wavelet Network. Our proposed algorithm [25, 26] allows the creation of a Deep Convolutional Wavelet Network (DCWN) for supervised image classification using:

- *The best contribution algorithm:* The best Contribution algorithm [15, 16] consists of learning a 2D wavelet network using the fast wavelet transform (FWT). The number of wavelets in the hidden layer evolves during learning until the desired value of Peak Signal to Noise Ratio (PSNR) is obtained. The main steps of the best contribution algorithm are [17]: Creation of the wavelet library (D), Decomposition of the 2D signal by FWT, Calculation of the contribution, Selection of the best wavelet and Reconstruction of the signal. With this

978-1-5090-2896-2/16/$31.00 ©2016 IEEE

algorithm, we tried to create a WN for every item of a class. Then, we calculate the score of wavelets using the number of its apparition in each position in the WN created previously to form a Global Wavelet Network (GWN) that approximates a unique class of the dataset.

- *A series of AEs:* The AE tries to learn an approximation to the identity function in a way that the output is similar to the input x [18, 19]. So, with the best wavelet selected in the begining, we apply a series of autoencoders to lead to a deep wavelet network (DWN) for the training of our base.
- *An intelligent pooling and a LCN:* The pooling layers are subsampled spatially, which reduces the spatial resolution of the representation and makes the representation vary smoothly with translations and small distortions of the input [20]. The Local Contrast Normalization (LCN) layer is inspired by computational neuroscience models [21]. This module performs local subtractive and divisive normalizations, enforcing a sort of local competition between adjacent features in a feature map, and between features at the same spatial location in different feature maps. So, for each layer of the DWN, we perform an intelligent pooling and a LCN to obtain a DCWN.
- *A linear classifier:* For the classification, we used a linear classifier like the Weak Learner [22]. A weak learner is defined to be a classifier which is only slightly correlated with the true classification.

The proposed DCWN tested with two different dataset (The Columbia Object Image Library (COIL-100) [23] and Arabic Printed Text Image (APTI) [24]) has shown a good performance in the classification test.

III. LEARNING ARCHITECTURES

In this section, we describe our approach used for feature extraction and learning. Our architecture does this by combining several techniques in a novel way.

We should recall that we make the classification of one class versus all classes from the dataset. So, we consider, every time, that the dataset consists of two classes: Class 1 and Class 2. Class 1 is the class that we want to classify. Class 2 contains all the other classes of the dataset.

Initially and as explained in [25, 26], we must create a WN for each image of Class 1 and Class 2 according to the best contribution algorithm (Fig. 1). After that, for all wavelets used in the WNs created in the previous step we compute their scores and we choose the best wavelet according to its score to create a Global Wavelet Network (GWN) that approximates one class (Class1).

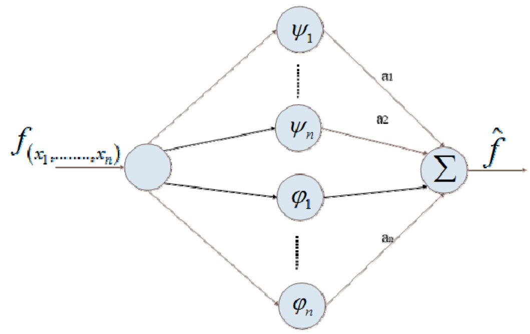


Fig. 1. Wavelet Network for one image.

To calculate the wavelets score, we should count the apparition number of every wavelet in every position in the WN for all the images from Class 1 and Class 2 separately (Fig. 2). In this way, we lead to the global score of the wavelets that approximate Class 1 and Class 2 using equation 1:

$$\underset{\text{Score in Class1}}{\psi_1} = \sum_{i=1}^{m} NumApp_i * ((\mathrm{m} - i) + 1)$$

$$and \tag{1}$$

$$\underset{\text{Score in Class2}}{\psi_1} = \sum_{i=1}^{m} NumApp_i * ((\mathrm{m} - i) + 1)$$

So, the global coefficient for ψ_1 is defined by:

$$\underset{\text{Global Score}}{\psi_1} = \underset{\text{Score in Class1}}{\psi_1} - \underset{\text{Score in Class2}}{\psi_1} \tag{2}$$

Wavelets Class 1

Position	ψ_1	ψ_2	ψ_3	ψ_4	...	ψ_n
1	5	0	30	100		0
2	60	10	3	0		20
3	9	0	60	3		0
4	0	22	0	35		88
⋮						
n	50	26	0	0		66

Wavelets Class 2

Position	ψ_1	ψ_2	ψ_3	ψ_4	...	ψ_n
1	0	50	3	10		100
2	40	40	0	20		0
3	69	0	0	54		22
4	20	0	10	3		8
⋮						
n	53	0	0	70		0

Fig. 2. Apparition number of the wavelets from Class 1 and Class 2.

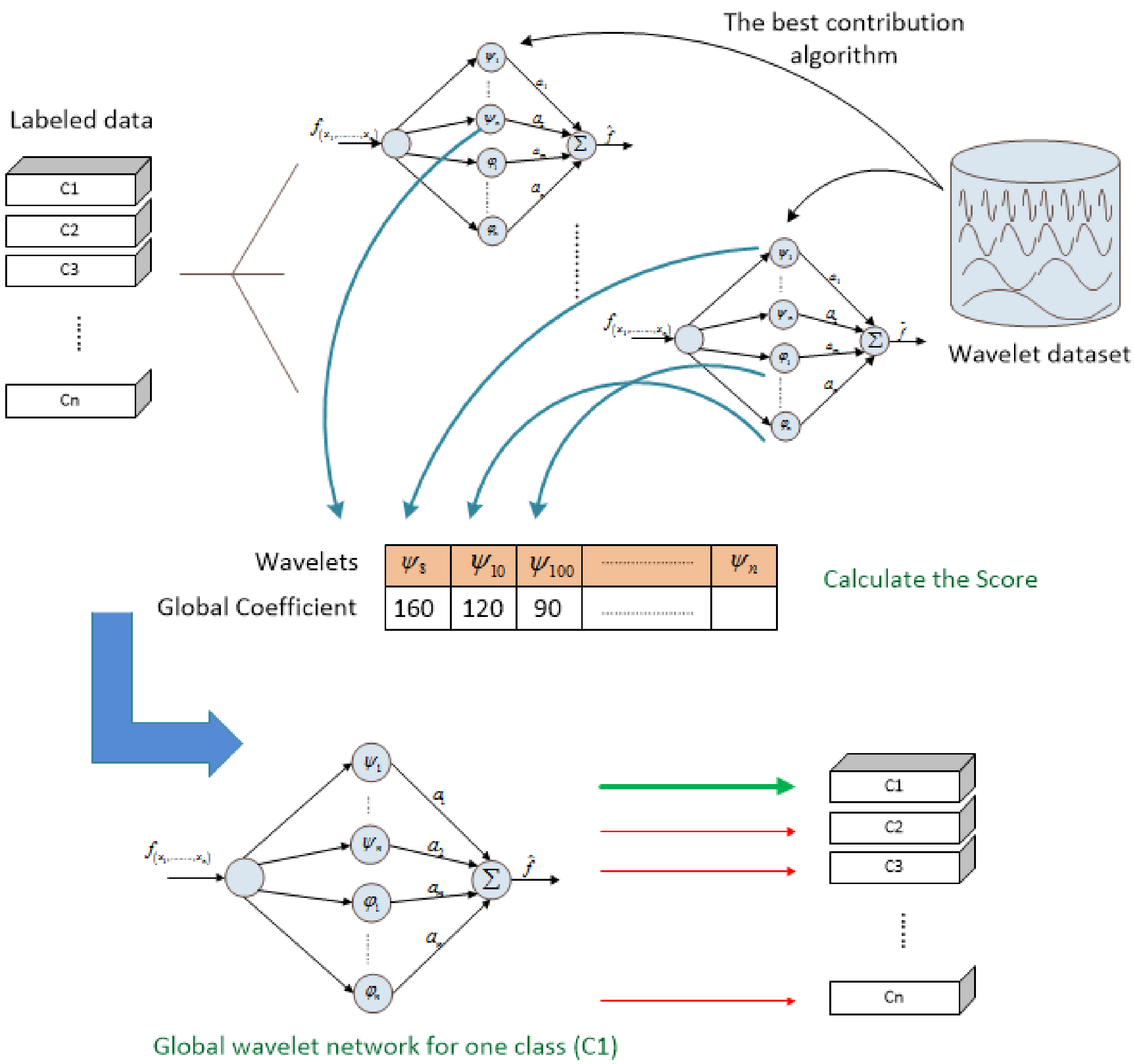


Fig. 3. Process to create Global Wavelet Network for one class.

Figure 3 illustrates the process of creation of our GWN that approximates one class begining by the WN for one image to obtains a GWN for all images from one class.

Our GWN is composed of a set of wavelets in the hidden layer noted ψ and φ whose ψ have wavelet details and φ are the wavelets for approximation. That is to say:

$$\hat{f}_{GWN} = \sum_{i=1}^{m} a_i \psi_i + \sum_{i=1}^{m} b_i \varphi_i \tag{3}$$

At this step, we use the concept of sparse coding to improve the performance of our GWN and to obtain a Global Sparse Wavelet Network (GSWN) that approximates one class from the dataset. As a first step, we create a new dataset that contains patches from the dataset of images and choose only the unique patches (Fig. 4).

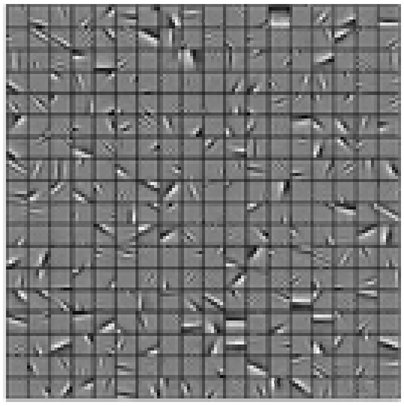


Fig. 4. Used base of patch.

Therefore, we begin to add a randomly chosen patch to our GWN. Then, we calculate the PSNR value. If it is improved, we keep the patch in our network, otherwise,we remove it. This step is stopped when reaching a defined error rate and we obtain finally a GSWN (Fig. 5).

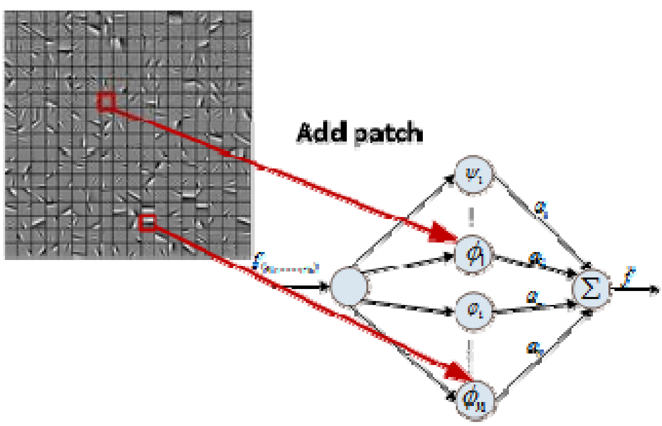


Fig. 5. Use of the patch in the GWN.

So, our algorithm is defined as follows:

Algorithm 1 Add patch to GWN

Input: data $\hat{f}$, data ϕ , val $psnr$

Initialize $errorRate = 0.75$

Repeat

$\phi_i = Random(\phi)$

Add ϕ_i To $\hat{f}$

$newPSNR = getPSNR()$

If $newPSNR > psnr$ **Then**

Remove ϕ_i from $\hat{f}$

Else

$psnr = newPSNR$

Endif

Until $newPSNR < errorRate$

Therefore, the signal of our GSWN is characterized as follows:

$$\hat{f}_{GSWN} = \sum_{i=1}^{m} a_i \psi_i + \sum_{i=1}^{m} b_i \varphi_i + \sum_{i=1}^{m} c_i \phi_i \qquad (4)$$

After the feature extraction with our GSWN and for the training phase, a series of stacked sparse autoencoders is applied. And for each hidden layer of our stacked autoencoders, we make an intelligent pooling [25, 26].

Then, we use a Softmax classifier at the last layer for our deep sparse wavelet network to test our approach (Fig. 6) considering that the Softmax is an activation function specialized for classification. Finally, the fine-tuning with the backpropagation algorithm is applied to all the hidden layers to greatly improve the performance of a stacked autoencoder.

IV. Experiment Results and Discussion

In this part, we describe the experimental test and we present some results. Our approach is tested on the ImageNet dataset (Fig. 7).

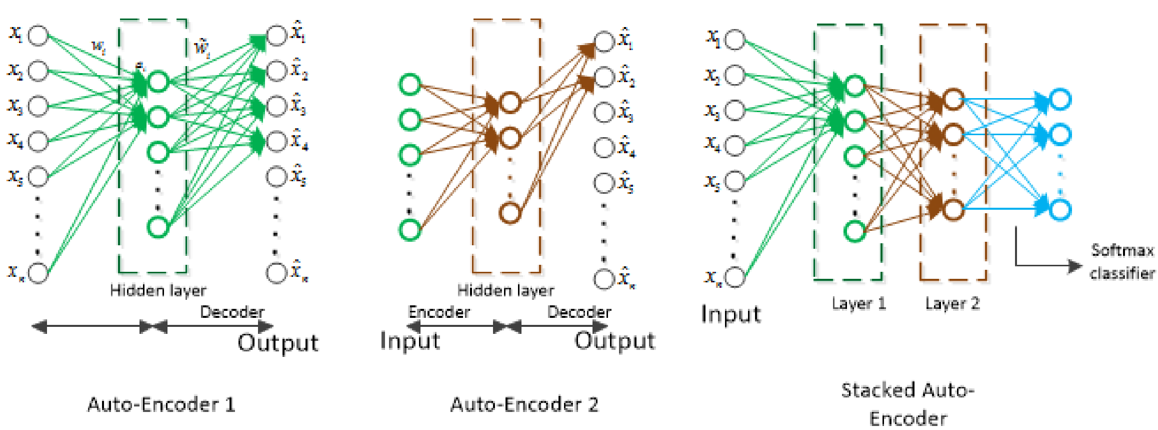


Fig. 6. Stacked Sparse Autoencoders.

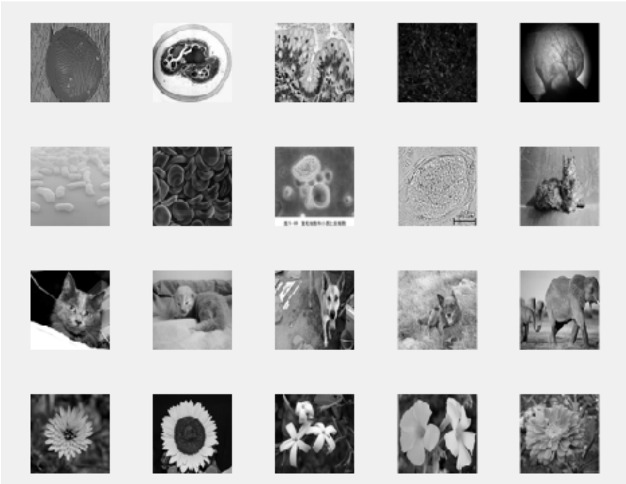
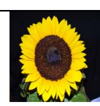
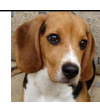


Fig. 7. Some images used from the ImageNet.

It is a set of images according to the wordNet hierarchy in which each node of the hierarchy is depicted by hundreds and thousands of images. The number of images used is 3000 where 2000 images for the training phase and 1000 images dedicate to the test phase.

The first result obtained by our approach is features selected from the GSWN. These features of each image demonstrate that the wavelets and the patches used in our GSWN approximate only one class from all others class from the dataset. In table I, we have PSNR values of some images from Class 1 and Class 2 tested with the GSWN for Class 1.

TABLE I. PSNR RESULT GIVEN BY THE GSWN OF EACH CLASS

Images from IMAGENET				
PSNR with GSWN that approximate it class	293.9714	291.0944	287.9055	289.8577
PSNR with GSWN for another Class	-12.5357	-20.1917	-22.2922	-16.6820

In the next step, we apply a series of a stacked autoencoders to learn our features. Our Deep Sparse Wavelet Network is obtained with two hidden layers (obtained with the stacked autoencoders) and a Softmax classifier in the last layer is represented in the figure 8.

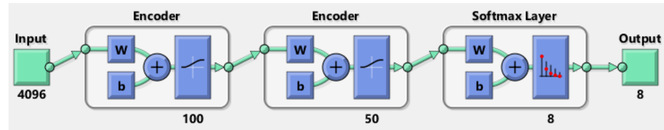


Fig. 8. An example of the Deep Sparse Wavelet network with 2 hidden layer and a Softmax classifier.

Applying the fine tuning with the backpropagation algorithm, we get the classification rate. With this result, we deduce the performance of our approach when the classification rate is compared with another approach for image classification tested on the ImageNet database (Table II).

TABLE II. THE IMAGE CLASSIFICATION RESULTS OF DIFFERENT METHODS ON THE IMAGENET DATABASE.

Methods	**Classification rates**
Our approach	91.0%
DCWN [25, 26]	86.4%
ILSVRC-2012 [27]	84.7%
VGG [28]	93.2%

V. CONCLUSION

This approach arises from the combination of the performance of the most used theories in the domain of image classification: The Wavelet Network and the Deep Learning. So, this approach permits the learning of a Sparse Wavelet Auto-Encoder through wavelets' network approximation. A group of Sparse Wavelet Auto-Encoders trained by this system are stacked to construct Deep Stacked Sparse Wavelet Auto-Encoders, then, a Softmax classifier is used in the last layer.

Despite all the good results tested on the ImageNet dataset described here, it is clear that this approach encourages us to propose it to other applications.

ACKNOWLEDGMENT

The authors would like to acknowledge the financial support of this work by grants from General Direction of Scientific Research (DGRST), Tunisia, under the ARUB program.

REFERENCES

[1] Y. Bengio., Learning deep architectures for ai. In Foundations and Trends in Machine Learning, pages 1-127, 2009.

[2] L. Deng and D. Yu., Deep Learning Methods and Applications. Foundations and Trends in Signal Processing, volume 7, 2014.

[3] S. Rifai, P. Vincent, X. Muller, X. Glorot and Y. Bengio, Contractive AutoEncoders: Explicit Invariance During Feature Extraction. in Proceedings of the 28th International Conference on Machine Learning, Bellevue, WA, USA, 2011.

[4] Y. LeCun, Learning invariant feature hierarchies. Computer Vision ECCV, 2012.

[5] X. Glorot and Y. Bengio, Understanding the di_culty of training deep feed-forward neural networks. In Proceedings of Artificial Intelligence and Statistics (AISTATS), 2010.

[6] C.-Y. Liou, W.-C. Cheng, J.-W. Liou and D.-R. Liou, Autoencoder for words. Neurocomputing, 139:8496, 2014.

[7] H. C. Shin, M. R. Orton, D. J. Collins, S. J. Doran and M. O. Leach, Stacked Autoencoders for Unsupervised Feature Learning and Multiple Organ Detection in a Pilot Study Using 4D Patient Data. IEEE Transactions on Pattern Analysis and Machine Intelligence, Volume 35, Issue 8, 2012.

[8] H. Lee, A. Battle, R. Raina and Y. Ng. Andrew, Efficient sparse coding algorithms. In Advances in neural information processing systems (pp. 801-808) 2006.

[9] B. A. Olshausen and D. J.Field, Sparse coding with an overcomplete basis set: A strategy employed by V1? Vision research, 37(23), 3311-3325, 1997.

[10] A. Coates and Y. Ng. Andrew, The importance of encoding versus training with sparse coding and vector quantization. In Proceedings of the 28th International Conference on Machine Learning (ICML-11) (pp. 921-928).

[11] S. Iyengar, E. Cho and V. Phoha, Foundations of wavelet networks and applications. Chapman and Hall/CRC Press, 2002.

[12] H. Szu, B. Telfer and S. Kadambe, Neural network adaptative wavelets for signal representation and classification. Optical Engineering, 31:19071961, 1992.

[13] Y. C. Pati and P. S. Krishnaprasad, Analysis and synthesis of feedforward neural networks using discrete affine wavelet transformations. IEEE Trans. Neural Networks, 4:7385, Jan 1993.

[14] Q. Zhang and A. Benveniste, Wavelet networks. IEEE Trans. On Neural Networks, 3(6):889898, 1992.

[15] M. Zaied, S. Said, O. Jemai and C. Ben. Amar, A novel approach for face recognition based on fast learning algorithm and wavelet network theory. Int. J. Wavelets, Multiresolution Inf. Process, 9(6):923945, 2011.

[16] O. Jemai, M. Zaied, C. Ben. Amar and A. M. Alimi, Fbwn: An architecture of fast beta wavelet networks for image classi_cation. In in Proceedings of the International Joint Conference on Neural Networks, 2010.

[17] A. Eladel, R. Ejbali, M. Zaied and C. Ben. Amar. A new system for image retrieval using beta wavelet network for descriptors and fuzzy decision. In in Proceedings of the International Conference of Soft Computing and Pattern Recognition, pp. 232-236, 2014.

[18] C. Y. Liou, W. C. Cheng, J. W. Liou, and D. R. Liou, Autoencoder for words. Int. J. Neurocomputing, 139:8496, 2014.

[19] Y. Bengio, Foundations and Trends in Machine Learning, Vol. 2. 2009.

[20] Y. LeCun, Learning invariant feature hierarchies. In in Proceedings of the ECCV, pp. 496505, 2012.

[21] N. Pinto, D. Cox and J. DiCarlo, Why is real-world visual object recognition hard? PLoS Computational Biology, 4(1), 2008.

[22] C.Tsai, W. Huang and M. Chiang, Mobile, ubiquitious and intelligent computing. Int. J. Lecture Notes in Electrical Engineering, 274:629636, 2014.

[23] S. Nene, S. Nayar and H. Murase. Columbia object image library (coil-100). Technical report, Columbia University, 1996.

[24] F. Slimane, S. Kanoun, J. Hennebert, A. M. Alimi, and R. Ingold. A study on font-family and font-size recognition applied to Arabic word images at ultra-low resolution. In Pattern Recognit. Lett., vol. 34, no. 2, pp. 209218, 2013.

[25] S. Hassairi, R. Ejbali and M. Zaied, A deep convolutional neural wavelet network to supervised arabic letter image classification. In Proceedings of the ISDA, pp. 496505. IEEE, 2015.

[26] S. Hassairi, R. Ejbali and M. Zaied, Supervised image classification using deep convolutional wavelets network. In in Proceedings of 27th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 265 - 271. IEEE, 2015.

[27] A. Krizhevsky, I. Sutskever and Geo_rey E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks. In in Proceedings of Advances in Neural Information Processing Systems 2012.

[28] K. Simonyan and A. Zisserman, Very deep convolutional networks for large-scale image recognition. In arXiv preprint arXiv:1409.1556 2014.