An empirical study: ELM in face matching

Haidong Wang[†], Md Fawwad Hussain[†], Himadri Mukherjee[‡], Sk Md Obaidullah[+], Ravindra S. Hegadi[#], Kaushik Roy[‡], K.C. Santosh[†⋆⋆], *IEEE* Senior Member

[†]Department of Computer Science, University of South Dakota
santosh.kc@ieee.org

[‡]Department of Computer Science, West Bengal State University
kaushik.mrg@gmail.com

[+]Department of Computer Science, Aliah University
sk.obaidullah@gmail.com

[#]Department of Computer Science, Solapur University
rshegadi@gmail.com

Abstract. Face data happens everywhere and face matching/verification is the must, such as it helps track criminals; unlock your mobile phone; and pay your bill without credit cards (e.g. Apple Pay). More often, in real world, grayscale image data are used since the color images require more storage. Gray level faces can be studied through two different features: edges and texture since spatial properties could be preserved. Such features could be used to classify faces from one another. Instead of using distance-based feature matching concept, in this paper, a fast machine learning classifier, which we call extreme learning machine (ELM) is used, where we have taken several different activation functions, such as *tanh*, *sigmoid*, *softlim*, *hardlim*, *gaussian*, *multiquadric* and *inv_multiquadric*. In our tests, five different publicly available datasets, such as Caltech, AR, ColorFERET, IndianFaces and ORL are used. For all activation functions, we have tested with and without feature selection techniques, and compared with the state-of-the-art results.

Keywords: Gray level images, spatial features, face matching, extreme learning machine, activation functions, feature selection.

1 Introduction

Face data happens everywhere and face matching/verification is the must, such as it helps track criminals; unlock your mobile phone; and pay your bill without credit cards (e.g. Apple Pay). Therefore, automated face recognition system is the need. Since 60's, there has been a remarkable success in this domain and commercial tools are available. However, we face a number of challenges in making an accurate and a robust face detection and recognition in real-world

⋆⋆ Corresponding author

environments. More often, uncontrolled environments can be one of the primary challenges. This includes variations in illumination, face pose and expression, just to name a few.

In face recognition, aforementioned difficulties cannot be sidelined. Even though human face varies from person to person, human face is not a unique rigid object and each of the faces have a variety of deformations. Therefore, other physical properties, such as gender, age and emotions can help add more information to create a challenge for the research scientist in computer vision.

In statistical pattern recognition, face matching/recognition can be made

a) either by feature-based matching, where similarity score (typically based on the distance function) can help decide unknown face images (test images) with the help of known ones.
b) or by feature-based training (using known face images) via machine learning classifier(s), where decision can be made for unknown face images (test images).

The paper is the complement of the work presented earlier [13], where the idea of feature-based matching has been strictly followed. Such a technique suffers from the high time complexity issue, since test face image(s) require(s) to match with all known face images in the database. We are aware that one would be interested in recognizing unknown faces in a limited amount of time. Not only time, memory can be considered as another issue for a compact (and/or reasonable) face matching tool/technique. Therefore, in this paper, using the exact same features that are employed in [13], we basically rely on training machine learning classier for a decision making.

The primary goal of our research is to build an automated face recognition systems with the use of the fast machine learning classifier. With this idea, we employ an Extreme Learning Machine (ELM) – a well known fast machine learning classifier in the domain, so decision can quickly be made unlike the conventional feature matching-based concept (via distance function). In the ELM, we study several different activation functions, such as *tanh*, *sigmoid*, *softlim*, *hardlim*, *gaussian*, *multiquadric* and *inv_multiquadric* to check which one performs the best in terms of recognition rate and processing time. Besides, we employ feature selection techniques to check whether we can improve the performance of the system in terms of decision making time and memory in use. Like before [13], we also hold the similar concepts of using grayscale images instead of color ones: A few of the reasons can be summarized as follows: a) more often, video surveillances are used to store grayscale images because of its storage issue; b) edge information can be considered as an important feature for applications, such as line-rich object detection (face image, for instance) and such an attribute can be achieved in gray scale images; and c) having both issues, it is always interesting to use grayscale images by realizing tools with low computational complexities and overheads.

The remainder of the paper can be organized as follows. Section 2 provides related works. In Section 3, we clearly explain the proposed work. Section **??**,

datasets, evaluation metrics and results (with analysis) are provided. Section **??** concludes the paper.

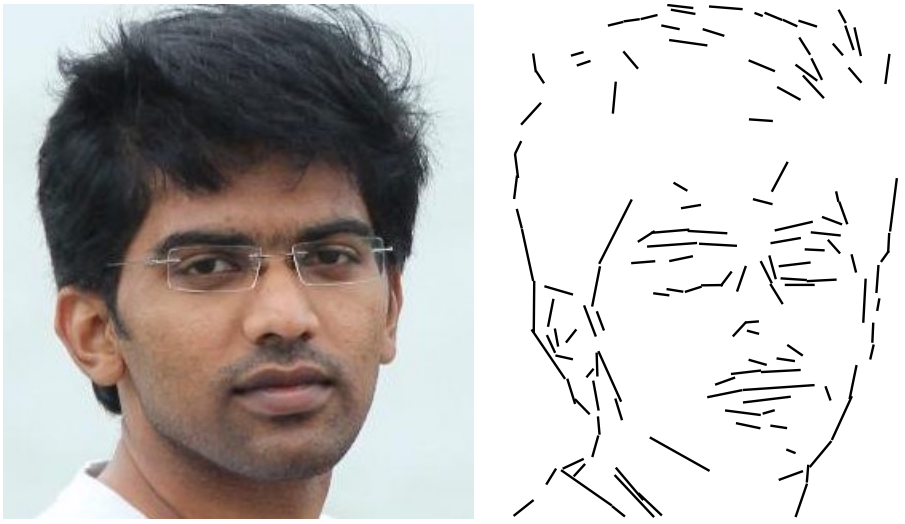
2 Related works

Many different approaches and techniques have been developed so far in the field of face matching and recognition for the reliable and efficient system. More often, face matching uses appearance-based concept, where they consider shape descriptors are rich enough to describe its appearance. However, they depend on the complexity of the problem, and in a few cases, all pixels are equally important that leads to high computational complexities and requires a high degree of correlation between known and unknown face images. In case when face images are taken from the controlled environment, not all pixels (from the images) are required to recognize i.e., a few of them (but selected ones) are sufficient. In all respects, training machine learning classifiers can help decide unknown face images like humans do. Training can be quick and is desirable in case extracted features are compact.

As reported earlier, since this study uses the exact same features from another work [13], in this section, we particularly focused on shape-based object description. It is desirable to have a descriptor that includes non-dense image information, such as edges or various key points and/or regions so they are able to cope deformations and variations in pose, scale, orientation as well. Since a decade, line-rich objects have been described by the set of line segments i.e., edges in general. To have an idea of this, we refer to the work reported in 2015 [5], where they present a rotation-and-scale-invariant, line-based color-aware (RSILC) descriptor for face matching in terms of their key-lines, line-region properties, and line spatial arrangements. The concept was inspired from previous work reported in 2014 [17]. Authors claimed that their descriptor is color-aware, invariant to rotation/scale and is robust to illumination changes. In 2002, line edge map (LEM) was introduced that is able to extract features from the line segments of a face image [9], which is found to be robust and efficient as compared to Eigenface [18]. We also noted that curve edges collect more information [7], and it is called by the name curve edge map.

Key lines (including edges i.e. edge map) may not take texture information into account. Since texture is an important attribute, in computer vision, Local Binary Pattern (LBP) [1] proved to be a powerful descriptor for face recognition. We find that face recognition using the LBP performs better in terms of speed and recognition rate. It can also handle face images with different facial expressions, different lighting conditions, image rotation and aging.

Integrating both key line (can be edge map, see Fig. 1) with texture features could potentially produce better performance in object recognition.



a) b)

Fig. 1. An example of an a) input Image; and its corresponding b) edge based representation.

3 Contributions

In Fig. 1, one can observe that edge map i.e. key line features, alone, can help identify human face. However, for comprehensive object recognition, texture-based features can complement key key line-based features. This means that we would like to express the importance of spatial features in our study: face matching. Following our work [**?**], the following features are considered.

a) For edge map-based features, we employ pyramid histogram of orientation gradients (PHOG) i.e., histogram of edge orientations and spatial distribution of edges [2]. More importantly, both descriptors do not suffer from memory issue. To extract texture-based features, we employ low level representation of the shape called the spatial envelope [15], which encodes spatial frequencies and orientations.
b) Once we have features in place, we train extreme learning machine and for decision making, we use several different activation functions, such as *tanh*, *sigmoid*, *softlim*, *hardlim*, *gaussian*, *multiquadric* and *inv_multiquadric*. For all activation functions, we test the ELM with and without feature selection techniques, and compared with the state-of-the-art results.

3.1 Spatial features

For more detailed information about spatial features for face representation and recognition, we refer to [**?**].

For PHOG, we worked on [2], since it captures the spatial distribution of edges and stored them as 1D vector representation. PHOG is inspired by the two sources: the image pyramid representation of Lazebink et al, [14] and the Histogram of Gradient Orientation (HOG) of Dalal and Triggs [6]. Local appearance can be described by a histogram of edge orientations (quantized into K bins) within an image subregion. The edge orientations are quantized into K bins, each of which represents the number of edges which have a certain angular range orientations. The spatial layout of the shape is based on the concept of spatial pyramid matching [10]. Each image is divided into a sequence of increasingly finer spatial grids by doubling the number of grids in each axes direction. The number of points in each grid cell is then recorded. The PHOG descriptor of the entire image is a vector with dimensionality $K \times \sum_{l \in L} 4^l$. For example, for level L = 3 and K = 20 bins, the size of the PHOG feature vector is of $(20 \times (4^0 + 4^1 + 4^2 + 4^3))$ 1700.

Regarding gist: spatial envelope, we worked on Oliva and Torralba [15] so it produces a very low dimensional representation of the image, which we call spatial envelope. Spatial envelope can be represented by a low-dimensional vector that encodes the distribution of orientations and scales in the image along with a coarse description of the spatial layout. The spatial envelope representation that has semantic attributes about the image provides a way of computing high-level image and space similarities between 2D sequences/images. In our study, for an input face image, a gist descriptor is computed by: a) convolving the image with 32 Gabor filters at 4 scales, 8 orientations, producing 32 feature maps of the same size of the input image; b) dividing each feature map into 16 regions (by a 4×4 grid); and c) concatenating the 16 averaged values of all 32 feature maps. As a result, it produces a feature vector of size $16 \times 32 = 512$.

3.2 ELM and activation functions

Extreme learning machines [8,11,12,4] are feedforward neural networks for a variety of challenges, such as classification, regression, clustering, sparse approximation, compression and feature learning with a single layer or multiple layers of hidden nodes, where the parameters of hidden nodes (not just the weights connecting inputs to hidden nodes) need not be tuned. Note that this is not the first time, we work on ELM for face matching [19]. Compared to [19], in our study, we used spatial features. Also, as different activation function provides different results, we use the following.

a) Tanh: It is a shifted as well as scaled variant of the logistic sigmoid transfer function. It depicts asymptotic symmetry and also leads to faster convergence. In general, it can be expressed as, $tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.
b) Multiquadric and inverse multiquadratic: The multiquadric function is a type of radial basis function. It is neither a positive definite nor a reproducing activation function. It is conditionally negative definite in nature and it produces a unique solution. In general, we write it as, $multiquadric(x) = \sqrt{1 + x^2}$. Therefore, we have $inv\_multiquadric(x) = \frac{1}{\sqrt{1+x^2}}$.

c) Hardlim and softlim: It is the short form for hard limit transfer function. This function outputs 1 when the net input value reaches a threshold and 0 in the other condition. Using python package, we express hardlim as $hardlim(x) = numpy.array(x > 0)$. Inversely, we have $softlim(x) = numpy.clip(x, 0.0, 1.0)$.
d) Sigmoid: This function derives its name from its "S" like shape and hence the name sigmoid. It is a kind of logistic function whose domain is all real numbers. It returns monotonically increasing values mostly between 0 to 1, and simply it can be expressed as, $sigmoid(x) = \frac{1}{1+e^{-x}}$.
e) Gaussian: It is a bell shaped continuous function. Its output is interpreted in terms of class membership which depends on the closeness of a value to a threshold. In general, it can be written as, $gaussian(x) = e^{-x^2}$.

3.3 Feature selection

Instead of relying on complete features extracted from the data, it is wise to use select them so that important and/or distinguished attributes are preserved for recognition. With this concept in mind, we employ the following technique [16].

Linear models penalized with the L1-norm have sparse solutions: many of their estimated coefficients are zero. When the goal is to reduce the dimensionality of the data to use with another classifier, they can be used along with feature selection (L1 based) to select the non-zero coefficient. In particular, sparse estimators useful for this purpose are the lasso for regression, and of Logistic regression and SVM for classification. With SVMs and logistic-regression, the parameter C controls the sparsity: the smaller C the fewer features selected. With Lasso, the higher the alpha parameter, the fewer features selected. In short, L1 regularization adds a penalty $\sum_i^n |W|$ to the loss function (L1-norm). Since each non-zero coefficient adds to the penalty, it forces weak features to have zero as coefficients. Thus L1 regularization produces sparse solutions, inherently performing feature selection. For more detailed information, please visit https://scikit-learn.org/stable/modules/feature_selection.html.

4 Results

4.1 Datasets and evaluation metric

For a through test, it is highly recommended to use standard datasets (publicly available) so that fair comparison is possible. In our study, the following datasets are used:

a) Caltech dataset;
b) AR face dataset;
c) Color FERET dataset;
d) ORL dataset; and
e) Indian face dataset.

Table 1. Results on Caltech dataset: average recognition rate (in %) and time (in seconds).

Activation function	Avg.	Avg. (FS)	Time	Time (FS)
tanh	0.988064386	0.994570991	0:00:14.930470	0:00:10.031956
sigmoid	0.986382294	0.990485189	0:00:14.902907	0:00:09.932660
softlim	0.984676056	0.994560776	0:00:14.478697	0:00:09.727814
gaussian	0.988643863	0.995020429	0:00:14.832572	0:00:09.955706
multiquadric	0.988635815	0.99546476	0:00:14.440570	0:00:09.771844
inv_multiquadric	0.989199195	0.9936619	0:00:14.523050	0:00:09.720032
hardlim	0.959702213	0.971470889	0:00:15.010304	0:00:09.646865

Table 2. Results on AR dataset: average recognition rate (in %) and time (in seconds).

Activation function	Avg.	Avg. (FS)	Time	Time (FS)
tanh	0.885454545	0.927636364	0:00:19.683048	0:00:14.822395
sigmoid	0.878636364	0.940727273	0:00:19.314931	0:00:14.650711
softlim	0.830454545	0.882545455	0:00:18.740531	0:00:14.296371
gaussian	0.859090909	0.912	0:00:19.012984	0:00:14.693428
multiquadric	0.893181818	0.930909091	0:00:18.686136	0:00:14.327555
inv_multiquadric	0.872272727	0.908727273	0:00:18.743095	0:00:14.414840
hardlim	0.624090909	0.596	0:00:18.754820	0:00:14.383245

The California Institute of Technology (Caltech) is a world-renowned science and engineering research and education institution. It has an important part in discovering new knowledge and innovations. For face recognition system, the dataset provided by Caltech contains 450 frontal face images of 27 distinct subjects with varying conditions. The AR Face dataset was developed by Aleix Martinez and Robert Benavente in the Computer Vision Center (CVC) at the U.A.B. The rich dataset contains over 3000 face images corresponding to 126 subject which includes 70 men and 56 women. The FERET database was developed under the supervision of Face Recognition Technology (FERET) program to develop new techniques, technology, and algorithms for the automatic recognition of human faces. For FR system, the dataset consists of 500 frontal face images of 105 distinct subjects with varying conditions. In our study, we ignore color pixels. The ORL database of faces was developed under the supervision of Cambridge University Laboratory. The database consists of 400 frontal face images of 40 distinct subjects with varying lighting conditions, poses and facial expressions. The Indian face database was developed by IIT Kanpur. There are eleven different face images of each of 40 distinct subjects. Different orientations of the face are included for example looking front, looking left, looking right, looking up, looking up looking down along with different facial emotions.

For our validation, we performed k-fold cross validation, where $k = 5$, and reported the average results (see sections below).

Table 3. Results on ColorFERET dataset: average recognition rate (in %) and time (in seconds).

Activation function	Avg.	Avg. (FS)	Time	Time (FS)
tanh	0.957460317	0.980302521	0:00:07.979022	0:00:03.684192
sigmoid	0.955873016	0.970957983	0:00:08.009312	0:00:03.586650
softlim	0.95	0.980302521	0:00:07.924479	0:00:03.478982
gaussian	0.945767196	0.979092437	0:00:07.991454	0:00:03.575078
multiquadric	0.96031746	0.975697479	0:00:07.806041	0:00:03.457342
inv_multiquadric	0.96031746	0.981445378	0:00:07.736843	0:00:03.481617
hardlim	0.895661376	0.942857143	0:00:07.706740	0:00:03.555562

Table 4. Results on Indian faces dataset: average recognition rate (in %) and time (in seconds).

Activation function	Avg.	Avg. (FS)	Time	Time (FS)
tanh	0.667037037	0.692740741	0:00:24.481315	0:00:22.178926
sigmoid	0.654444444	0.691851852	0:00:24.500733	0:00:21.841826
softlim	0.638888889	0.655407407	0:00:23.904217	0:00:21.502019
gaussian	0.651851852	0.675555556	0:00:24.265890	0:00:21.955657
multiquadric	0.668888889	0.689481481	0:00:23.812052	0:00:21.522418
inv_multiquadric	0.657407407	0.670518519	0:00:23.836929	0:00:21.512923
hardlim	0.537037037	0.52562963	0:00:23.925644	0:00:21.694868

Table 5. Results on ORL dataset: average recognition rate (in %) and time (in seconds).

Activation function	Avg.	Avg. (FS)	Time	Time (FS)
tanh	0.964375	0.99	0:00:13.672983	0:00:09.582739
sigmoid	0.965625	0.9855	0:00:13.605367	0:00:09.427775
softlim	0.971875	0.9895	0:00:13.729829	0:00:09.161826
gaussian	0.966875	0.993	0:00:13.587183	0:00:09.431332
multiquadric	0.97375	0.99	0:00:13.402822	0:00:09.337323
inv_multiquadric	0.970625	0.9925	0:00:13.519410	0:00:09.224058
hardlim	0.951875	0.9715	0:00:13.433424	0:00:09.239333

Table 6. Feature dimensions with and without FS.

Dataset	Features (dimension) without FS	with FS
Caltech	2212	577
AR	2212	977
ColorFeret	2212	1192
Indian	2212	1298
ORL	2212	802

4.2 Results, analysis and comparison

In Tables 1 to 5, one can see the following:

a) Tests were done separately for different datasets.
b) For all datasets, tests were compared with and without Feature Selection (FS). Note that for all datasets, we have a feature vector of size 2212 i.e., 1700 (PHOG) + 512 (gist).
c) For all activation functions, both recognition rates (in %) and processing times (in seconds) were provided. For simplicity, note that average results were reported (using k-fold cross validation, where k=5).

With these in hands, we observe the following:

a) We have achieved recognition rate up to 99.50% (Caltech dataset). Indian face dataset is found to be difficult one as compared to others. Note that the proposed study is limited to whether i) activation functions need to be used correct and ii) feature selection can help.
b) Feature selection technique improved the performance with a maximum difference of 6.2% in accuracy and 5.632 seconds in processing time. The reason behind the positive jump is due to the fact that feature dimensions were reduced significantly, where redundancies were deleted (see Table 6).

5 Conclusion and future works

In this paper, we have studied different activation functions used in Extreme Learning Machine (ELM) by considering spatial features for gray level face image representation and recognition. Besides, we have demonstrated that the usefulness of the feature selection in both recognition rate and processing time. In our tests, we have compared (on five different publicly available datasets, such as Caltech, AR, ColorFERET, IndianFaces and ORL) activation functions used in ELM in addition to feature selection. In a new future, we plan to work on machine learning classifier-based idea, such as active learning [3] at the time when we need real data or live data.

References

1. Ahonen, T., Hadid, A., Pietikainen, M.: Face description with local binary patterns: Application to face recognition. IEEE transactions on pattern analysis and machine intelligence 28(12), 2037–2041 (2006)
2. Bosch, A., Zisserman, A., Munoz, X.: Representing shape with a spatial pyramid kernel. In: Proceedings of the 6th ACM international conference on Image and video retrieval. pp. 401–408. ACM (2007)
3. Bouguelia, M., Nowaczyk, S., Santosh, K.C., Verikas, A.: Agreeing to disagree: active learning with noisy labels without crowdsourcing. Int. J. Machine Learning & Cybernetics 9(8), 1307–1319 (2018)
4. Cambria, E., Huang, G., Kasun, L.L.C., Zhou, H., Vong, C., Lin, J., Yin, J., Cai, Z., Liu, Q., Li, K., Leung, V.C.M., Feng, L., Ong, Y., Lim, M., Akusok, A., Lendasse, A., Corona, F., Nian, R., Miche, Y., Gastaldo, P., Zunino, R., Decherchi, S., Yang, X., Mao, K., Oh, B., Jeon, J., Toh, K., Teoh, A.B.J., Kim, J., Yu, H., Chen, Y., Liu, J.: Extreme learning machines. IEEE Intelligent Systems 28(6), 30–59 (2013), https://doi.org/10.1109/MIS.2013.140
5. Candemir, S., Borovikov, E., Santosh, K., Antani, S., Thoma, G.: Rsilc: Rotation- and scale-invariant, line-based color-aware descriptor. Image and Vision Computing 42, 1–12 (2015)
6. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on. vol. 1, pp. 886–893. IEEE (2005)
7. Deboeverie, F., Veelaert, P., Philips, W.: Face analysis using curve edge maps. In: International Conference on Image Analysis and Processing. pp. 109–118. Springer (2011)
8. Ding, S., Zhao, H., Zhang, Y., Xu, X., Nie, R.: Extreme learning machine: Algorithm, theory and applications. Artif. Intell. Rev. 44(1), 103–115 (Jun 2015), http://dx.doi.org/10.1007/s10462-013-9405-z
9. Gao, Y., Leung, M.K.: Face recognition using line edge map. IEEE transactions on pattern analysis and machine intelligence 24(6), 764–779 (2002)
10. Grauman, K., Darrell, T.: The pyramid match kernel: Discriminative classification with sets of image features. In: Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on. vol. 2, pp. 1458–1465. IEEE (2005)
11. Huang, G., Huang, G.B., Song, S., You, K.: Trends in extreme learning machines. Neural Netw. 61(C), 32–48 (Jan 2015), http://dx.doi.org/10.1016/j.neunet.2014.10.001
12. Huang, G., Wang, D., Lan, Y.: Extreme learning machines: a survey. Int. J. Machine Learning & Cybernetics 2(2), 107–122 (2011), https://doi.org/10.1007/s13042-011-0019-y
13. Hussain, M.F., Wang, H., Santosh, K.: Gray level face recognition using spatial features. In: Selected papers, RTIP2R, Communications in Computer and Information Science. vol. 862, pp. –. Springer (2019)
14. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In: Computer vision and pattern recognition, 2006 IEEE computer society conference on. vol. 2, pp. 2169–2178. IEEE (2006)
15. Oliva, A., Torralba, A.: Modeling the shape of the scene: A holistic representation of the spatial envelope. International journal of computer vision 42(3), 145–175 (2001)

16. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830 (2011)
17. Santosh, K.C., Lamiroy, B., Wendling, L.: Integrating vocabulary clustering with spatial relations for symbol recognition. Int. J. Document Analysis & Recognition 17(1), 61–78 (2014)
18. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In: Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on. pp. 586–591. IEEE (1991)
19. Zong, W., Huang, G.: Face recognition based on extreme learning machine. Neurocomputing 74(16), 2541–2551 (2011), https://doi.org/10.1016/j.neucom.2010.12.041