Further Parameters Estimation of Neocognitron Neural Network Modification with FFT Convolution

D. Kangin[1], G. Kolev[2], A. Vikhoreva[3]
[1]*Department of informatics and control systems,*
Bauman Moscow State Technical University, 5 2-nd Baumanskaya Street, 117624, Moscow, Russia
[2] *Center of Coordination of International Scientific Education Programmes*
3B, Kul'neva Street, 121170, Moscow, Russia
[3] *Institute of Physics of the Earth, Russian Academy of Science*
dkangin@gmail.com

***Abstract* — This paper presents further development of an improved version of the neocognitron algorithm introduced by Fukushima. Some comparisons of other symbol recognition methods based on the neocognitron neural network are also performed, which led to the proposal of several modifications — namely, layer dimension adjustment, threshold function and connection Gaussian kernel estimation. The width and height are taken into account independently in order to improve the recognition of patterns of slightly different dimensions. The learning and recognition calculations are performed as FFT convolutions in order to utilize external specialized computing system. Finally, more detailed results of the neocognitron performance evaluation are provided.**

***Index Terms* — neocognitron, vehicle plate, recognition, neural networks**

I. Introduction

This paper proposes further results on development and parameter estimation of an improved version [1] of the neocognitron algorithm introduced by Fukushima [2].

The neocognitron neural network [2], [3], [4] is widely used for symbol recognition. This article describes the neocognitron neural network modification applied to vehicle plate symbols recognition. The state-of-the-art methods give the possibility of robust recognition, as it is described in this article, but they still need in further improvements in order to rise up time and recognition rate. The proposed improvements give the possibility of increase of both these characteristics.

The architecture of the neocognitron corresponds to one of the standard neocognitron variants [2], albeit with some changes introduced to the calculation algorithms and is comprised of layers of *S*-cells (simple cells) and *C*-cells (complex cells). In order to improve the performance of the neocognitron applied to vehicle plate symbols, some modifications were carried out:

- the threshold function and connection Gaussian kernel estimation methods were proposed;
- the method of layer dimensions adjustment was developed;
- independent width and height pattern dimensions were introduced;
- learning and recognition calculations were performed as FFT convolutions.

In addition, *S*-cells are used for feature extraction and are modified through unsupervised learning line-extracting cells and competitive supervised learning on the highest stages. In this process, *S*- and *C*-cell staggering is preserved.

This paper is organized as follows. Section II describes the state-of-the-art, giving a short review of existing methods. Section III performs a brief explanation of the neocognitron architecture, described accurately in [2]. Section IV gives a way of Gaussian kernel estimation for the neocognitron connections. Section V describes the threshold function estimation for *S*-layer output. Section VI is announcing the test set size choose method. Section VII describes the adjustment of the layers size for the test set patterns. Section VIII reports about the way of FFT convolution application in the neocognitron. Section IX describes the test and training patterns sets and the experimental evaluations. In conclusion, we summarize the results of the work and outline further directions of the research.

II. Neocognitron Comparison with Other Recognition Methods

At present, there are many vehicle plate symbol recognition methods in use, some of which are briefly compared here.

A very popular method, based on single-layer neural networks has recognition speed as a key advantage. However, as it requires the data set to be linearly separable, it is restricted to use with very limited class of input sets.

Multi-layer perceptron[5] can represent non-linear functional mappings. It provides a wide class of pattern sets due to its capacity to overcome constraints of linear separability. Although this network is widely used in practical recognition systems, as its main application is in solving general recognition problems, it does not take into consideration image patterns features. Some problems in selecting feature basis for recognition system should also be noted. For example, the feature basis should meet the requirements of sufficiently low noncorrelatedness and completeness. It can be based on statistical characteristics of input patterns, data in spatial and Fourier space, or be combined with dimension reduction methods. The additional requirement of this method is the ability to cope with input pattern transforms (shift, rotation, scale and perspective) and input signal noise of sufficiently low magnitude, which can be met by specific tuning of neural network or by using the specialized neural networks.

Cognitron [6] is a neural network similar to the visual cortex structure. It was one of first specialized neural networks developed and applied to image pattern recognition. However, it is not widely used in contemporary systems due to strong re-

strictions placed upon input pattern recognition (shift, rotation, scale and perspective).

Topological recognition methods are widely used in image pattern recognition [7], [8]. In common cases, topological methods are based on describing the equivalence of objects that can be mapped one to one using a continuous function. This approach allows for rotation, translation, and perspective transform independent pattern recognition. Thus, it can be implemented as an independent recognition method or be a part of the complex recognition system. One of the key tools of topological recognition is the homology group of simplicial or cubic complexes.

However, there are some theoretical problems, caused by straightforward application of this method, the key one being that this method is not persistent to noise. In order to overcome this limitation, simplicial complex calculation method introduces some improvements independent of noise impact, one of which is persistent homology groups [9]. The disadvantage of this method is that it yields rather unstable results on low-dimension data is used as input.

III. Neocognitron Architecture

The neocognitron neural network consists of several layers. The input layer U_0 contains one cell plane, used to input the pattern. The purpose of the next layer — a layer of contrast extracting cells U_G, containing two cell planes with concentric and off-centre receptive fields is to extract the contrast from the images independently of the mean of brightness. The output of the layer U_G serves as the input for the first *S*-layer — the first of four stages, each containing one *S*-and one *C*-cell layer — which is trained to extract edge component with different rotation angles. The first layer is used for feature extraction, while the second is used for blurring the *S*-layer output. The connections of *S*-cells at stages U_{S2} and U_{S3} are trained by unsupervised competitive learning as described in [2], [3], [10], whereas U_{S4} layer is trained by supervised competitive learning according to [2] and is used to show the recognition results.

IV. Gaussian Kernel Estimation

Both fixed and variable connections (a_{Cl} , c_{Sl} and a_{Sl}, b_{Sl}, respectively) should be estimated as Gaussian function. The Gaussian kernel can be estimated by various methods. The method described in this work is based on overall correlation estimation. Denoting the pattern pixel intensity as the chance quantity (0, 0), we can hypothesize, that the neighbouring pixel, shifted on (*x, y*) relatively to the pattern pixel, has correlated chance quantity ξ (*x, y*). Thus, we can calculate the Pearson product-moment correlation coefficient between ξ(0, 0) and ξ (*x, y*) as:

$$\rho(\xi(0,0),\xi(x,y)) = \frac{\sum_{i=1}^{n}(\xi_i(0,0)-\mu_\xi(0,0))(\xi_i(x,y)-\mu_\xi(x,y))}{\sqrt{\sum_{i=1}^{n}(\xi_i(0,0)-\mu_\xi(0,0))^2\sum_{i=1}^{n}(\xi_i(x,y)-\mu_\xi(x,y))^2}}, \quad (1)$$

where summation index *i* runs over all of the pixels of the pattern images. Consequently, the resulting function can be approximated by Gaussian, applying the least squares method. The connectivity range is chosen by the three-sigma rule.

V. Threshold Function Estimation

S-layer output threshold value θ_l selection is sufficient for learning and recognition. Our aim is to maximize the output value u_{Sl} on the seed cell:

$$u_{S_l}(\vec{n},\vec{k}) \to \max. \quad (2)$$

Thus we can find the value of the θ_l threshold, so that u_{Sl} achieves the maximum value.

S-layer output is determined as

$$u_{S_l}(\vec{n},\vec{k}) = \frac{\theta_l}{1-\theta_l}\times \varphi\left[\frac{1+\sum_{\kappa=1}^{K_{C_{l-1}}}\sum_{|\vec{v}|<A_{S_l}} a_{S_l}(\vec{v},\kappa,\vec{k})u_{C_{l-1}}(\vec{n}+\vec{v},\kappa)}{1+\theta_l b_{S_l}(\vec{k})v_l(\vec{n})}-1\right], \quad (3)$$

$$v_l(\vec{n}) = \sqrt{\sum_{\kappa=1}^{K_{C_{l-1}}}\sum_{|\vec{v}|<A_{S_l}} c_{S_l}(\vec{v})\left[u_{C_{l-1}}(\vec{n}+\vec{v},\kappa)\right]^2}, \quad (4)$$

where $a_{S_l}(\vec{v},\kappa,\vec{k})$ — variable *C*-layer connection weight value, $b_{Sl}(k)$ — variable lateral inhibition, θ_l — positive S-layer threshold, $c_{S_l}(\vec{v})$ — fixed weight, $\vec{n}$ — the receptive field centre, $\vec{v}$ — the radius of the summation range, $\varphi(x) = \max(0, x)$.

If we have the fixed count of learning iterations M, we can write

$$a_{S_l}(\vec{v},\kappa,k) = a^0_{S_l}(\vec{v},\kappa,k) + Mq_l c_{S_l}(\vec{v})u_{C_{l-1}}(\vec{n}+\vec{v},\kappa). \quad (5)$$

According to Fermat's theorem, the equation should be solved as

$$\frac{\partial u_{S_l}}{\partial t} = 0. \quad (6)$$

Let

$$\alpha = 1+\sum_{\kappa=1}^{K_{C_{l-1}}}\sum_{|\vec{v}|<A_{S_l}} a_{S_l}(\vec{v},\kappa,k)u_{C_{l-1}}(\vec{n}+\vec{v},\kappa), \quad (7)$$

$$\beta = b_{S_l}(k)v_l(\vec{n}). \quad (8)$$

Then

$$u_{S_l}(\vec{n},k)=\frac{\theta_l}{1-\theta_l}\varphi\left(\frac{\alpha}{1+\theta_l\beta}-1\right). \tag{9}$$

When $1+\theta_l\beta>0$,

$$u_{S_l}(\vec{n},k)=\frac{\theta_l\alpha}{(1-\theta_l)(1+\theta_l\beta)}-\frac{\theta_l}{1-\theta_l}, \tag{10}$$

The derivative is then given by

$$\frac{\partial u_{S_l}(\vec{n},k)}{\partial\theta_l}=\frac{\beta(\alpha-\beta)\theta_l^2-2\beta\theta_l+\alpha-1}{(1-\theta_l)^2(1+\theta_l\beta)^2}. \tag{11}$$

As the result, we obtain

$$\theta_l=\frac{1-\sqrt{1-\frac{(\alpha-1)(\alpha-\beta)}{\beta}}}{\alpha-\beta}. \tag{12}$$

With that, we have estimated a value of θ_l threshold.

VI. Choosing the Experimental Sample Set

The approximation of the experimental sample set can be found according to the central limit theorem (CLT), which states that the sum set of independent events (in this case learning result) that are independent and equally distributed, converges to normal distribution. Thus, CLT can be applied to the test set, which can be approximated as the set of equally distributed random quantities:

$$\begin{aligned}&\forall u_0^i : E\left[u_0^i\right]=\mu, D\left[u_0^i\right]=\sigma^2 \Rightarrow\\ &\Rightarrow E\left[\sum_{i=0}^{n}\frac{u_0^i}{n}\right]=\mu, D\left[\sum_{i=0}^{n}\frac{u_0^i}{n}\right]=\frac{\sigma^2}{n}.\end{aligned} \tag{13}$$

VII. Neocognitron Layer Dimensions Adjustment

The proposed version of the neocognitron enables independent adjustments to the neocognitron layers' width and height. Here, some heuristic formulae for neocognitron layer adjustment were applied, generalizing the Fukushima's neocognitron parametrization. Thus, the first layer U_0 dimensions are set according to input data dimensions, whereas the layer U_G dimensions are set according to the following expressions:

$$\begin{aligned}U_{GX}&=U_{0X}+A_{GX}-1,\\ U_{GY}&=U_{0Y}+A_{GY}-1,\end{aligned} \tag{14}$$

Such U_G dimension parameterization ensures, that all nonzero elements of the convolution matrix for input and connection matrices will be taken into account. The U_{C1}, U_{C2} and U_{C3} layers dimensions are set according to the following:

$$U_{CX_n}=2\left(\left\lfloor\frac{\left\lfloor\frac{U_{SX_{n-1}}}{2}\right\rfloor-\left\lfloor\frac{A_{CX_{n-1}}}{2}\right\rfloor+1}{m_x}\right\rfloor+\left\lfloor\frac{A_{CX_n}}{m_x}\right\rfloor\right)+U_{SX_{n-1}}\bmod 2-1, \tag{15}$$

$$U_{CY_n}=2\left(\left\lfloor\frac{\left\lfloor\frac{U_{SY_{n-1}}}{2}\right\rfloor-\left\lfloor\frac{A_{CY_{n-1}}}{2}\right\rfloor+1}{m_y}\right\rfloor+\left\lfloor\frac{A_{CY_n}}{m_y}\right\rfloor\right)+U_{SY_{n-1}}\bmod 2-1, \tag{16}$$

Here m_x and m_y denote cells thinning-out coefficients. These formulae describe the dimension reduction, enabled by the cell thinning-out procedure. They further ensure, that the parity of the layer dimensions is changing from layer to layer according to the half-pitch staggering, applied in neocognitron.

The U_{S2} and U_{S3} layers dimensions are set according to formulae

$$U_{SX_2}=U_{CX_1}+1, \tag{17}$$

$$U_{SY_2}=U_{CY_1}+1. \tag{18}$$

These expressions describe the half-pitch staggering, that requires the dimension parity change.

The U_{S1} layer dimension is given by the formulae:

$$U_{SX_1}=U_{GX}+1-\left\lfloor\frac{A_{SX_1}}{2}\right\rfloor-\left(\left\lfloor\frac{A_{SX_1}}{2}\right\rfloor\bmod 2\right), \tag{19}$$

$$U_{SY_1}=U_{GY}+1-\left\lfloor\frac{A_{SY_1}}{2}\right\rfloor-\left(\left\lfloor\frac{A_{SY_1}}{2}\right\rfloor\bmod 2\right). \tag{20}$$

These expressions take into account half-pitch staggering.

U_{S4} layer dimensions are calculated by formulae

$$U_{SX_4}=U_{CX_3}+A_{SX_4}, \tag{21}$$

$$U_{SY_4}=U_{CY_3}+A_{SY_4}, \tag{22}$$

U_{C4} layer dimensions are always set to 1:

$$U_{CX_4}=U_{CY_4}=1. \tag{23}$$

VIII. Layer Output Calculation Using FFT

The proposed version of neocognitron is based on the application of FFT [11] for input and connection convolutions on each layer. It allows for the convolution calculations to be

performed by the outer device performing FFT, thus increasing the performance.
The sequence of the convolution processing consists of the following steps:
1. input and connection matrices FFT calculation;
2. element-wise input and connection matrices FFT multiplication;
3. obtaining the resulting matrix reverse FFT;
4. potential for cells thinning-out, performed over the matrix, obtained on the previous step.
The layer output calculation can be regarded as the convolution of the layer input and connections, as is common approach in different signal processing applications. Such interpretation gives us the ability to use the convolution theorem:

$$F(f \otimes h) = F(f) \times F(h), \tag{24}$$

where operator F denotes the discrete Fourier transform (DFT), operator $\otimes$ is convolution, operator $\times$ refers to element-wise multiplication, and f, h — input and connections matrices.
In this work, the discrete convolution theorem is used, whereby the convolution is assumed to be cyclic, so that certain restrictions are applied to input and connection data. To exclude impact of the error, introduced by cyclic convolution, we set the size of convolution, connection and input matrices to

$$N_{0x} = N_{ix} + N_{cx} - 1, \tag{25}$$

$$N_{0y} = N_{iy} + N_{cy} - 1, \tag{26}$$

where N_{ix}, N_{iy} are dimensions of the required output matrix, N_{cx}, N_{cy} denote dimensions of the connection matrix. In order to meet these restrictions, both input and connection matrices are filled by zeros.
The next step is the element-wise input and FFT multiplication of connection matrices:

$$F(o) = F(f) \times F(h). \tag{27}$$

The result of the operation is processed by reverse FFT:

$$o = F^{-1}(F(o)). \tag{28}$$

If the cells thinning-out operation is performed, it should be applied to matrix o, resulting in dimensionality reduction (fig. 1).

IX. Experimental evaluations

The developed method evaluations were performed using one learning and one recognition set, each containing 5000 patterns, i.e. images of various sizes up to 14 x 17 pixels, which depict the Russian vehicle plate digits. Although the recognition rate is dependent on the training patterns, the proposed recognition demonstrated up to 96% recognition success rate, whereas the learning set yielded 100% of recognition. The examples of the learning and test pattern lists are shown on fig. 2.

Table 1 depicts comparison of the recognition rate for the proposed version of the neocognitron, the Fukushima's neocognitron with equal width and height, and state-of the-art method, using multi-layer perceptron. Also the time rate comparison is shown. As we can see, the proposed version of the neocognitron gives us a possibility to increase both the recognition rate by more exact fitting the layers' dimensions to patterns sizes and the time rate using FFT besides of matrix multiplication.
The recognition rate and time significantly depend on θ_{Ri} parameters. These parameters denote the threshold values θ_i, described in previous sections, for recognition (as opposed to θ_{Li} thresholds, used for learning), regulating the selectivity of *S*-cells, that is, the count of non-zero outputs of *S*-cells. The fig. 3-7 depict the dependency of the neocognitron on the threshold values. As can be seen on the fig. 3, the recognition rate dramatically decrease depending on θ_{Rl}. When the θ_{Rl} is sufficiently lower than the extremum, the threshold results in non-zero output for insignificant features of input pattern, so that the network is not robust to noise, while the values, higher than the extremum, give zero-output to most of the features, even significant. Both cases result in recognition rate decrease. The recognition rate graphs, depicted on the figures 5, 7, can be explained in a similar way, but here the low-threshold recognition rate decrease is not so noticeable. It is caused by lowering the input pattern noise impact for the higher layers. Figures 4, 6 depict the decrease of the recognition time with threshold decrease. It is caused by dependency of *S*-layers' planes count: the greater threshold we choose, the less planes we have.

X. Conclusion

The neocognitron neural networks have many parameters, implying the recognition results. Gaussian kernel estimation allowed us to achieve the sizes of the connectivity area on each layer in order to take into account correlations between neighbouring cells. Threshold function estimation gives us the possibility to control whether the feature is relevant to the input pattern. Independent width and height layer dimensions enable us to fit the sizes of layers to sizes of the input patterns. Using FFT, we can improve the speed of the neocognitron network, because it reduces the matrix multiplication to the operation of element-wise multiplication.
The evaluations of the proposed method were performed using one learning and one recognition set, each containing 5000 patterns. Patterns were presented by low-resolution images of various sizes up to 14 x 17 pixels, depicting Russian vehicle plate digits. The proposed modifications of the parameterization and evaluation of the neocognitron neural network gave us the possibility to improve both time from 1.06 to 1.09 ms and recognition rate of the neocognitron from 94.6 to 96 percent.
The parameters adjustment gives us a possibility for further improvement of the neural network algorithm and architecture. However, it is supposed that, in order to improve the recognition rate, some experiments for determination of neocognitron half-pitch staggering impact are needed. In addition, future work in this field should compare suggested convolution methods with existing alternatives, using other discrete

transforms, such as Hartley transform. One more direction that should be explored in more depth is the different ways of edge extraction layer learning, i.e., methods of generation of edge extraction layer patterns. The pattern preprocessing requirements should also be studied. Although neocognitron can perform the recognition of gray-scale images, the improvement of neocognitron performance depends of preprocessing algorithms. Heuristically, high-contrast images should give better results, when compared to low-contrast ones. However, the preprocessing method should be chosen according to more stringent arguments. Looking forward, it is also planned to analyze all the tasks required to perform video surveillance. In order for a proper symbol recognition pattern to be produced, a proper auto, license plate, and symbol segmentation should be performed. These tasks are currently under consideration, noting that some steps could be achieved via more complex algorithms, involving the neocognitron. For instance, using the selective attention mechanism, proposed by Fukushima, we could realize both symbol extraction and recognition in one algorithm. Thus, the task of neocognitron application as a part of more complex algorithm is being considered for our future work.

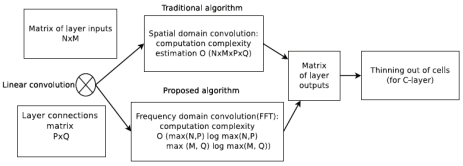

Figure 1: Cells thinning-out after multiplication.

Table 1
Comparison of the proposed neocognitron,
Fukushima's neocognitron and state-of-the-art multilayer perceptron

	Proposed neocognitron	Fukushima's neocognitron	Multi-layer perceptron
Average time for pattern, ms (learning patterns)	1.06	1.17	1.25
Rate, percent (learning patterns)	100	100	100
Average time for pattern, ms (test patterns)	1.09	1.21	1.31
Rate, percent (test patterns)	96.0	94.6	92.7

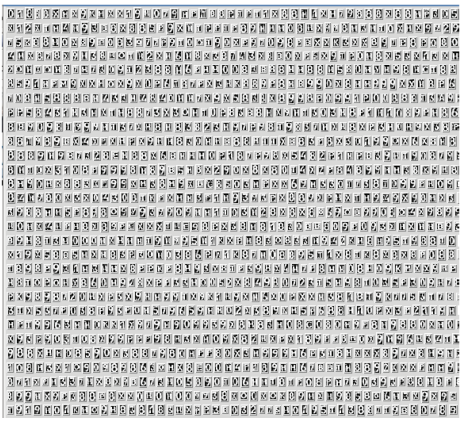

Figure 2: Learning and test pattern examples.

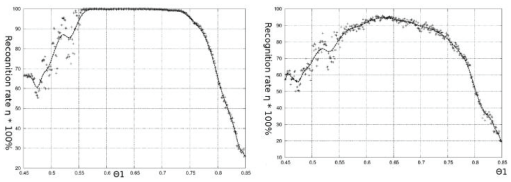

Figure 3: Recognition rate graphs for learning and test sets depending on θ_{R1}

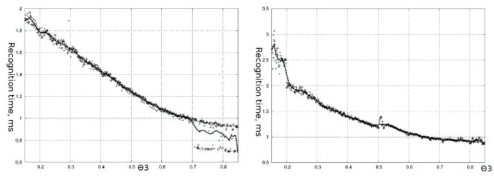

Figure 4: Recognition time graphs for learning and test sets depending on θ_{R3}

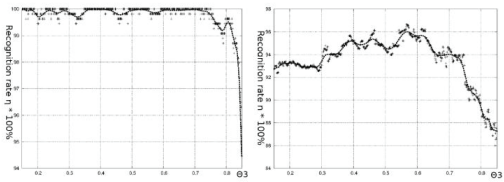

Figure 5: Recognition rate graphs for learning and test sets depending on θ_{R3}.

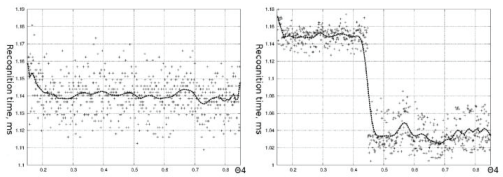

Figure 6: Recognition time graphs for learning and test sets depending on θ_{R4}

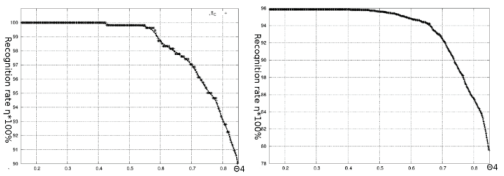

Figure 7: Recognition rate graphs for learning and test sets depending on θ_{R4}.

REFERENCES

[1] G. Kolev, D. Kangin (2012). Neocognitron neural network modi_cation with FFT convolution for vehicle plate symbols recognition, 2012 (Preprint)

[2] K. Fukushima (2003). Neocognitron for handwritten digit recognition. In Eckmiller, R. & Von der Malsburg, C. eds. Neurocomputing 51 (2003): 161-180.

[3] K. Fukushima (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biological Cybernetics, 36(4): 93-202.

[4] K. Fukushima (2011). Increasing robustness against background noise: Visual pattern recognition by a neocognitron. Neural networks 24 (7): 767-778.

[5] Bishop, C. M. (2006). Pattern Recognition and Machine Learning. Singapore: Springer, pp.225-290.

[6] K. Fukushima (1981). Cognitron: A selforganizing multilayer neural network model. NHK Technical Monograph No. 30, pp. 1-25.

[7] A. Dutta, S. Chaudhury(1993). Bengali Alpha-Numeric Character Recognition Using Curvature Features. Pattern Recognition 26, 17071720

[8] F. Kimura, M. Shridhar (1991) Handwritten Numerical Recognition Based on Multiple Algorithms. Pattern Recognition, Vol. 24, No. 10, pp. 969-983.

[9] G. Carlsson, A. Zomorodian, A. Collins, and L. Guibas. Persistence Barcodes for Shapes. International Journal of Shape Modeling, 11 (2005), 149-187.

[10] T. Kohonen (2001). Self-organizing maps. Berlin, Heidelberg, New York, Barcelona, Hong Kong, London, Milan, Paris, Singapore: Springer.

[11] William L. Briggs,Van Emden Henson (1995). The DFT: an owner's manual for the discrete Fourier transform. Society of industrial and applied mathematics, PA, USA: pp. 143-179.