DISCRETE ORTHOGONAL TRANSFORMS AND NEURAL NETWORKS FOR IMAGE INTERPOLATION

Miloš ORAVEC, Jaroslav POLEC
Department of Telecommunications FEI STU Bratislava
Ilkovičova 3, 812 19 Bratislava, Slovakia
Tel.: +421/7/68279 409, Fax: +421/7/68279 601
E-mail: oravec@ktl.elf.stuba.sk

Abstract

In this contribution we present transform and neural network approaches to the interpolation of images. From transform point of view, the principles from [1] are modified for 1st and 2nd order interpolation. We present several new interpolation discrete orthogonal transforms. From neural network point of view, we present interpolation possibilities of multilayer perceptrons. We use various configurations of neural networks for 1st and 2nd order interpolation. The results are compared by means of tables.

Keywords:

interpolation, discrete orthogonal transform, multilayer perceptron

1. Discrete Fourier Transform

1.1 One-dimensional interpolation with zero completing

The interpolation feature of 1D DFT is fairly known and described in basic textbooks about image processing [4]. The algorithm is clear from (1) [1].

$$X(k)=\frac{1}{N}\sum_{n=0}^{N-1}x(n).e^{-j\frac{2\pi kn}{N}},\ k=0,1,\ldots,N\text{-}1,$$

$$Y(l)=X(l),$$

for $l=0,1,\ldots,\frac{N}{2}-1$,

$$Y\left(BN-\frac{N}{2}+l\right)=X\left(\frac{N}{2}+l\right),$$

for $l=1,2,\ldots,\frac{N}{2}-1,$

$$Y(l)=0,$$

for $l=\frac{N}{2}+1,\frac{N}{2}+2,\ldots,BN-\frac{N}{2}-1$,

and $Y\left(\frac{N}{2}\right)=X\left(BN-\frac{N}{2}\right)=\frac{1}{2}X\left(\frac{N}{2}\right)$ (1)

Then

$$y(m)=B\sum_{l=0}^{BN-1}Y(l).e^{j\frac{2\pi lm}{BN}},$$

for $m=0,1,\ldots,BN-1$.

1.2 Zero interleaving and filtration interpolation

The spectrum mirroring principle of a zero interleaved signal and a following lowpass filtration is shown in [4]. In a higher order subsampling, this scheme can be considered as one step of a multistage algorithm. In 2D case, we have more ways of zero interleaving, but mainly the one from [5], where we speak about the n-th order interpolation. Interpolations (subsampling) of 1st and 2nd order are the most important for us.

2. Discrete cas-cas Transform (DCCT)

2.1 Interpolation 1D DHYT with zero completing

The process is very similar to DFT interpolation, only interpolating function is identical with DHYT basis. Consequently, it is necessary to alter only initializing and terminating step in (1).

2.2 Interpolation with zero interleaving DCCT

The spectrum can be described by the equation

$$\mathbf{Y}_N=\begin{pmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B} & \mathbf{A}\end{pmatrix} \quad (2)$$

where $\mathbf{Y}_N$ is a matrix with dimension NxN and **A**, **B** are matrices of dimension $N/2$x$N/2$.

Let us have the following transfer function of filter, which expresses the zero completing for image of half area. For image 8x8, **type 1** is

$$\begin{pmatrix} 1 & 1 & 1 & 0,5 & 0,5 & 0,5 & 1 & 1 \\ 1 & 1 & 1 & 0,5 & 0,5 & 0,5 & 1 & 1 \\ 1 & 1 & 0,5 & 0 & 0 & 0 & 0,5 & 1 \\ 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 & 0,5 \\ 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 & 0,5 \\ 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 & 0,5 \\ 1 & 1 & 0,5 & 0 & 0 & 0 & 0,5 & 1 \\ 1 & 1 & 1 & 0,5 & 0,5 & 0,5 & 1 & 1 \end{pmatrix} \tag{3}$$

As a reconstructed image, we will call an image produced by following operation:

$$\mathbf{y}_{N_1 x N_2} = 2 \cdot \left(\mathbf{U}_{N_1}\right)^T \cdot \left(\mathbf{H}_{N_1 x N_2} \circ \mathbf{Y}_{N_1 x N_2}\right) \cdot \mathbf{U}_{N_2} \tag{4}$$

where

$$\mathbf{Y} = \mathbf{H} \circ \mathbf{Y} \tag{5}$$

is so called Hadamard product, for which

$$Y_{ij} = H_{ij} \circ Y_{ij} \ . \tag{6}$$

2.3 Interpolation with zero completing with DCCT 2

The spectrum can be described by the formula

$$\mathbf{Y}_N = \begin{pmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{J}_{N/2} \cdot \mathbf{B} \cdot \mathbf{J}_{N/2} & \mathbf{J}_{N/2} \cdot \mathbf{A} \cdot \mathbf{J}_{N/2} \end{pmatrix} \tag{7}$$

The transfer function of filter **type 2** is the matrix

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0,5 & 0,5 & 0,5 \\ 1 & 1 & 1 & 1 & 1 & 0,5 & 0,5 & 0,5 \\ 1 & 1 & 1 & 1 & 1 & 0,5 & 0,5 & 0,5 \\ 1 & 1 & 1 & 0,5 & 0,5 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0,5 & 0,5 & 0 & 0 & 0 \\ 0,5 & 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 0,5 & 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 0,5 & 0,5 & 0,5 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{8}$$

The transfer function of filter **type 3** is the matrix, which for image 8x8 is

$$\mathbf{H}_8 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{9}$$

Or, after taking a non-separable spectrum symmetry (around reverted diagonal) in account, we can have also another type of filter transfer function, which is good with rectangular transforms. We will call it the **type 4** transfer function.

$$\mathbf{H}_8 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0,5 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0,5 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0,5 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0,5 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0,5 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0,5 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0,5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0,5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \tag{10}$$

3. 2D Interpolation of 1st Order - Conclusion

From the preceding parts and other experiences with 2D DOTs, we can derive Table 1.

4. 2D interpolation of 2nd order (eventually even B_1, B_2 order)

4.1 2D interpolation of 2nd order with zero completing

In this case, an image is separable subsampled, consequently the interpolation can be done separately: zero completing as in 1D case for the row spectrum and for the spectrum of transformed and completed rows in spectrum after column transform.

2D DOT MIRRORING	DCT II	DCCT	DFT	DCCT2	DST	MHT	DCT III	DLT	HT
TYPE	no	1	1	2	2	2	2	no	no
FILTER TYPE	-	1	1	2	3,4	3,4	3,4	-	-

Tab. 1 Interpolation qualities of DOTs in transform of zero interleaved signal (1st order subsampling)

4.2 Zero interleaving interpolation

Spectrum of transforms from Tab. 1, which have feature of mirroring the spectra of image zero interleaved, is for image $\hat{\mathbf{x}}_{N_1 x N_2}$

A. 2D DFT and DCCT

$$\mathbf{Y}_{N_1 \times N_2} = \begin{pmatrix} \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} & \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} \\ \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} & \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} \end{pmatrix} \qquad (11)$$

B. 2D DST, MHT, WST, IDCT II and IDLT

$$\mathbf{Y}_{N_1 x N_2} = \begin{pmatrix} \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} & \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} \cdot \mathbf{J}_{\left(\frac{N_2}{2}\right)} \\ \mathbf{J}_{\left(\frac{N_1}{2}\right)} \cdot \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} & \mathbf{J}_{\left(\frac{N_1}{2}\right)} \cdot \mathbf{A}_{\left(\frac{N_1}{2}\right)\times\left(\frac{N_2}{2}\right)} \cdot \mathbf{J}_{\left(\frac{N_2}{2}\right)} \end{pmatrix} \qquad (12)$$

Then

$$\mathbf{y}_{N_1 x N_2} = 4 \cdot \left(\mathbf{U}_{N_1}\right)^T \cdot \left(\mathbf{H}_{N_1 \times N_2} \circ \mathbf{Y}_{N_1 \times N_2}\right) \cdot \mathbf{U}_{N_2} \qquad (13)$$

Results for image NELA of size 256x256 pixels are presented in Tab. 2 and 3.

2D DOT	overlap		
	8x8/8x8 MSE	8x8/4x4 MSE	16x16/8x8 MSE
DFT, DCCT	42,941	17,747	18,075
DCT III	47,208	26,301	18,741
DST	55,433	38,339	20,617
MHT	68,362	47,587	
DLT	82,017	43,223	
WST	42,628	37,28	

Tab. 2 Dependence of MSE between 1st order interpolated image and original from different DOTs, block size and overlap size, where nominator is size of transform block and denominator is size of block retained as a result of interpolation.

2D DOT	overlap		
	8x8/8x8 MSE	8x8/4x4 MSE	24x24/8x8 MSE
DFT, DCCT	166,953	52,546	46,473
DCT III	1074,254	259,975	64,453
DST	1444,731	574,661	107,157
MHT	1960,868	886,458	
DLT	1502,382	547,572	
WST	168,396	169,453	

Tab. 3 Dependence of MSE between 2nd order interpolated image and original from different DOTs, block size and overlap size, where nominator is size of transform block and denominator is size of block retained as a result of interpolation.

5. Multilayer Perceptron (MLP) as a Non-linear Interpolator

To perform neural network interpolation of the 1st and the 2nd order, we concentrate on the multilayer perceptron (MLP). The basic MLP building unit is the model of an artificial neuron. This unit computes the weighted sum of the inputs plus the bias weight and passes this sum through the activation function (usually sigmoid).

In a multilayer configuration, the outputs of the units in one layer form the inputs to the next layer. The weights of the network are usually computed by training the network using the backpropagation algorithm.

A multilayer perceptron represents nested sigmoidal scheme [6], [3] - its form for the single output neuron is

$$F(\mathbf{x}, \mathbf{w}) = \varphi\left(\sum_j w_{oj} \varphi\left(\sum_k w_{jk} \varphi\left(\ldots \varphi\left(\sum_i w_{li} x_i\right)\ldots\right)\right)\right) \qquad (14)$$

where $\varphi(\cdot)$ is a sigmoidal activation function, w_{oj} is the synaptic weight from neuron j in the last hidden layer to the single output neuron o, and so on for the other synaptic weights, x_i is the i -th element of the input vector $\mathbf{x}$. The weight vector $\mathbf{w}$ denotes the entire set of synaptic weights ordered by layer, then neurons in a layer, and then number in a neuron.

Here we use the standard backpropagation algorithm without any modifications. Our notation for the MLP architecture is following: for a 3-layer neural network, the configuration n-m-p means that the MLP contains n input neurons, m hidden neurons, and p output neurons.

We are interested in 2 approaches in using MLP as the 1st order interpolator:

- non-block approach - 1 pixel is interpolated from 4 surrounding pixels
- block approach - 4x4 block is taken, 8 missing values are reconstructed from 8 original pixels.

For 2nd order interpolation, we use block approach only; 4x4 block is taken, 12 missing values are reconstructed from 4 original pixels.

For training the neural networks, a 512x512 training image constructed from fragments of images of various faces and sizes was used. Lena 256x256 image (called here NELA) was used for testing.

The results for 1st order interpolation for non-block and block approaches are shown in Tab. 4 and 5, respectively.

The results for 2nd order interpolation for block approach are shown in Tab. 6.

From the presented tables it can be seen, that within one chosen method there is no significant difference between the chosen MLP architecture, i.e. the simplest architecture can be used. For 1st order interpolation, very fast training can be noticed (by 1 cycle we mean presenting training image to the network).

Network	MSE	PSNR [dB]	Number of cycles
4-1	16,82	35,87	5
4-2-1	16,5	35,96	3
4-8-1	16,97	35,83	1
4-16-1	17,65	35,66	3
4-8-16-1	17,24	35,76	6

Tab 4. 1st order interpolation results for NELA 256x256 by various MLP configurations, trained by standard backpropagation, learning rate parameter=0.1, shuffled patterns, non-block approach

Network	MSE	PSNR [dB]	Number of cycles
8-8	25,8	34,02	3
8-16-8	29,17	33,48	6
8-16-32-8	27,17	33,79	30

Tab 5. 1st order interpolation results for NELA 256x256 by various MLP configurations, trained by standard backpropagation, learning rate parameter=0.1, shuffled patterns, block approach (4x4 blocks)

Network	MSE	PSNR [dB]	Number of cycles
4-12	128,4	27,05	40
4-8-12	128,3	27,05	200
4-8-10-12	128	27,06	300

Tab 6. 2nd order interpolation results for NELA 256x256 by various MLP configurations, trained by standard backpropagation, learning rate parameter=0.001, shuffled patterns, block approach (4x4 blocks)

References:

[1] Wang,Z.,Wang,L.: Interpolation Using the Fast Discrete Sine Transform, Signal Processing, No.26, 1992, 131-137

[2] Sontag,E.D.: Remarks on Interpolation and Recognition Using Neural Nets, Advances in Neural Information Processing Systems, Vol. 3, 1991, 911-917

[3] Hornik,K.: Multilayer Feedforward Networks Are Universal Approximators, Neural Networks, Vol. 2, 1989, 359-365

[4] Kotuliaková, J., Petrík, T., Rozinaj, G.: Signály a sústavy, ES SVŠT, Bratislava, 1988

[5] Farkaš, P., Herrera, S., Tvarožek, M.: Adaptive Image Coding Based on Generalized IBTC, In: Bauerfeld, W., Spaniol, O.: Broadband Islands 1994, Hamburg, 147-154

[6] Haykin,S.: Neural Networks (A Comprehensive Foundation), Macmillan College Publishing Company, New York, 1994

About authors...

Jaroslav POLEC received the Ing. (MSc.) degree in electrical engineering, PhD. degree in telecommunications and Doc. degree from FEI STU in Bratislava. At present he is the associate professor at the Department of Telecommunications of FEI STU in Bratislava. His research interests include signal processing and probability models.

Miloš ORAVEC received the Ing. (MSc.) degree in electrical engineering and Dr. degree in applied informatics from FEI STU in Bratislava. At present he is lecturer at the Department of Telecommunications of FEI STU in Bratislava. His research interests include signal processing and neural networks.