DOI 10.37882/2223-2966.2024.11-2.35

# МАТЕМАТИЧЕСКАЯ МОДЕЛЬ И АЛГОРИТМЫ РАСПОЗНАВАНИЯ ПРИ ОБРАБОТКЕ ИЗОБРАЖЕНИЙ

## MATHEMATICAL MODEL AND RECOGNITION ALGORITHMS FOR IMAGE PROCESSING

***V. Chasovskikh***
***E. Koch***

*Summary*. The article discusses the possibility of increasing the speed of algorithms when processing information technology images, including artificial intelligence technologies. A mathematical model is proposed as a nonlinear tensor product. This model is presented in the form of fast algorithms. It has been proven that various types of fast Fourier transform can be represented in a nonlinear, generalized form and represented by a set of two rules defining nonlinearity for fast algorithms. For image processing algorithms, relationships are formed from several operations of fast algorithms. Three iterations are defined. In the first iteration, the nonlinearity of the transformation is determined. In the second iteration, the nonlinearity of sparse transformations of direct sums of nonlinear transformations is formed. At the third iteration, a fast nonlinearity transformation is formed as a result of combining nonlinear sparse transformations. A new approach to transformations with nonlinear dependencies and the use of a fast algorithm has been created. The applied recursion rules for nonlinear transformation models create new opportunities for regular viewing of implicit fast transformations with nonlinearity. Thus, the authors formulated a new property of transformations, defined as the result of a superposition of «sparse» transformations with nonlinearity, and a new (with fewer computational operations) fast algorithm was created.

*Keywords*: model, pixels, image, fast algorithms. information Technology.

***Часовских Виктор Петрович***
*Доктор технических наук, профессор, ФГБОУ ВО Уральский государственный экономический университет*
*u2007u@ya.ru*
***Кох Елена Викторовна***
*Кандидат сельскохозяйственных наук, доцент, ФГБОУ ВО Уральский государственный экономический университет*
*elenakox@mail.ru*

*Аннотация*. В статье рассматривается возможность повышения скорости алгоритмов при обработке изображений информационных технологий, включая технологии искусственного интеллекта. Предлагается математическая модель, как нелинейное тензорное произведение. Эта модель представлена в форме быстрых алгоритмов. Было доказано, что различные виды быстрого преобразования Фурье можно представить в нелинейном, обобщенном виде и представлять множеством из двух правил, определяющих нелинейность для быстрых алгоритмов. Для алгоритмов при обработке изображений формируются соотношения из нескольких операций быстрых алгоритмов. Определены три итерации. В первой итерации определяется нелинейность преобразования. Во второй итерации формируется нелинейность разреженных преобразований прямых сумм нелинейных преобразований. На третьей итерации формируется быстрое преобразование нелинейности как результат совмещения нелинейных разреженных преобразований. Создан новый подход к преобразованиям с нелинейными зависимостями и использованием быстрого алгоритма. Применяемые правила рекурсии для моделей нелинейных преобразований создают новые возможности для регулярного просмотра неявных быстрых преобразований с нелинейностью. Таким образом, авторы сформулировали новое свойство преобразований, определяемое как результат суперпозиции «разреженных» преобразований с нелинейностью, и создан новый (с меньшим количеством вычислительных операций) быстрый алгоритм.

*Ключевые слова*: модель, пиксели, изображение, быстрые алгоритмы. информационные технологии.

В современной практике информационных технологий нелинейный подход стал важным методом обработки транзакций преобразования и распознавания изображений (основная технология искусственного интеллекта). В настоящей практике доминируют алгоритмы компьютерной обработки больших данных средствами информационных технологий для выявления в них каких-либо зависимостей и закономерностей, а также определении частотных компонентов.

Например, при обработке данных видеокамер, алгоритмы выявляют их спектральные характеристики и после распознавания изображений, алгоритмами улучшают качество и плотности сжатия.

Целью работы является разработка модели, позволяющей существенно уменьшить количество необходимых операций в алгоритмах обработки изображений, создать так называемые быстрые алгоритмы.

## Материалы и методы исследования

При проведении исследования в качестве материалов использовались многоканальные изображения, применяемые в системах дистанционного зондирования Земли при решение различных научных и прикладных задач. Для получения достоверных результатов применялись методы математической статистики и теории вероятностей, метод статистических наблюдений и выборочных совокупностей, методы гиперкомплексных чисел для обработки изображений и распознавания объектов, присутствующих в этом изображение. Применение традиционных алгоритмов моделирования с помощью нелинейных фильтров определялось общим нелинейным интегралом свертки Вольтерра или G-функцией Винера [1, с. 440; 2, с. 178]. Авторы ориентировались на алгоритмы распознавания и обработки изображений в которых доминируют операции агрегирования [3, с. 390; 4, с. 401; 5, с. 68]. Авторы рассматривают единый подход к быстрым нелинейным преобразованиям (FNLT).

## Результаты исследования и их обсуждение

Для построения математических моделей и алгоритмов распознания и обработки изображений авторы применяют быстрое преобразование Фурье (Fast Fourier transform — FFT), подтвердившее высокую эффективность в подобных приложениях [1, с. 440; 2, с. 178].

Изображение, представляющее собой пиксельное множество, переводится в частотный ряд. Представление изображения в виде набора частот используется в целях их обработки и фильтрации. Оно позволяет применять различные фильтры, осуществлять устранение шумов, улучшать контраст, выделять границы и отдельные детали.

Такую операцию авторы обращают в пространство пикселей, что существенно упрощает процесс обработки изображений и позволяет выполнять сложные операции с данными (нелинейная обработка), с высокой эффективностью. Однако такое пространство пикселей, как объём исходной информации, влияет на количество необходимых вычислительных операций.

Для нелинейной обработки авторы предлагают две новые формы, основанные на последовательном разложении точечного (пиксельного) FFT. Предложенные формы основаны на том, что они создаются посредством ряда быстрых рекурсивных унитарных преобразований произвольного порядка [2, с. 178; 6, с. 60; 7, с. 80]. Авторы обобщают быстрые линейные преобразования на некоторые нелинейные случаи, определяемые параллельным/последовательным объединением элементарных нелинейных преобразований 2x2 (нелинейные отображения-бабочки) [8, с. 72; 9, с. 87]. Применяя эту структуру, авторы определяют большое семейство FFT и получаем ряд известных и новых результатов об алгоритмах FFT, существующих преобразованиях и определяем структурные свойства между преобразованиями.

Основной элемент-точечного $2^n$ FFT определяет операции со сложными данными — блок-бабочка (см. рис. 1): $\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} f_0(x_0,x_1) \\ f_1(x_0,x_1) \end{bmatrix}$. Бабочка имеет 2 входа и 2 выхода. Значения на входах называются $a_{i_1}$ и $b_{i_2}$, значения на выходах называются $c_{j1}$ и $d_{j2}$. Комплексное число $W$(коэффициент вращения) является весовым коэффициентом и различно для каждой бабочки. Комбинируя операции «бабочка» подходящим образом, создается 2N-точечное FFT (см. рис. 2). Предложенный алгоритм FFT-бабочки показывает поток данных и операции с данными в графической форме.

Авторы рассматривают простейшую версию *явно обратимого* преобразования: оно состоит из замены переменных с участием двух *произвольных* функций $g_0(\cdot,\cdot), g_1(\cdot,\cdot)$ от двух величин $x_1, x_2$ до двух величин $y_1, y_2$ и наоборот: $\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} g_0(\cdot_0,\cdot_1) \\ g_1(\cdot_0,\cdot_1) \end{bmatrix}\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} g_0(x_0,x_1) \\ g_1(x_0,x_1) \end{bmatrix}$. Если $g_0(\cdot_0,\cdot_1) = g_{00}(\cdot_0) + g_{01}(\cdot_1), g_1(\cdot_0,\cdot_1) = g_{10}(\cdot_0) - g_{11}(\cdot_1)$ то

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} g_{00}(\cdot_0) + g_{01}(\cdot_1) \\ g_{10}(\cdot_0) - g_{11}(\cdot_1) \end{bmatrix} \circ \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} =$$

$$= \begin{bmatrix} g_{00}(\cdot_0) & g_{01}(\cdot_1) \\ g_{10}(\cdot_0) & g_{11}(\cdot_1) \end{bmatrix} \circ \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} g_{00}(x_0) + g_{01}(x_1) \\ g_{10}(x_0) + g_{11}(x_1) \end{bmatrix}.$$

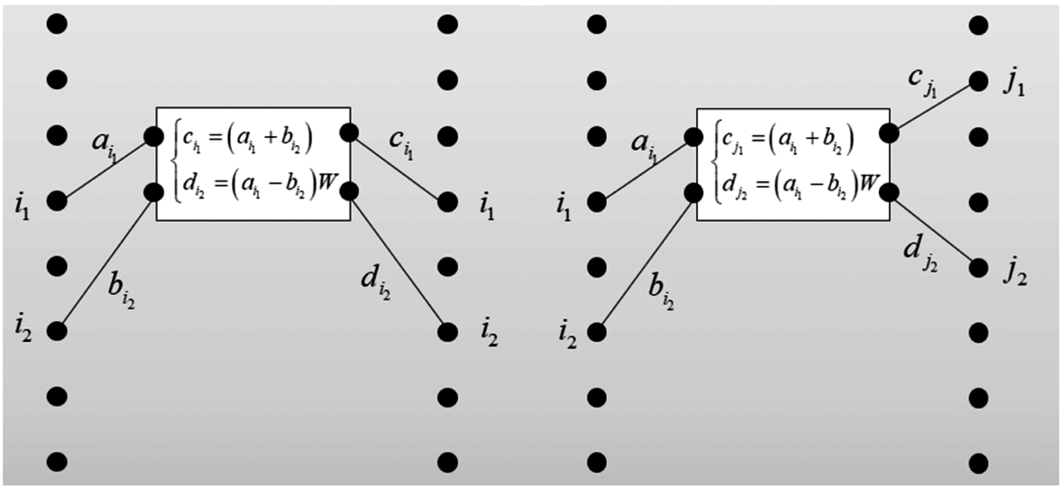


Рис. 1. Неместные операции

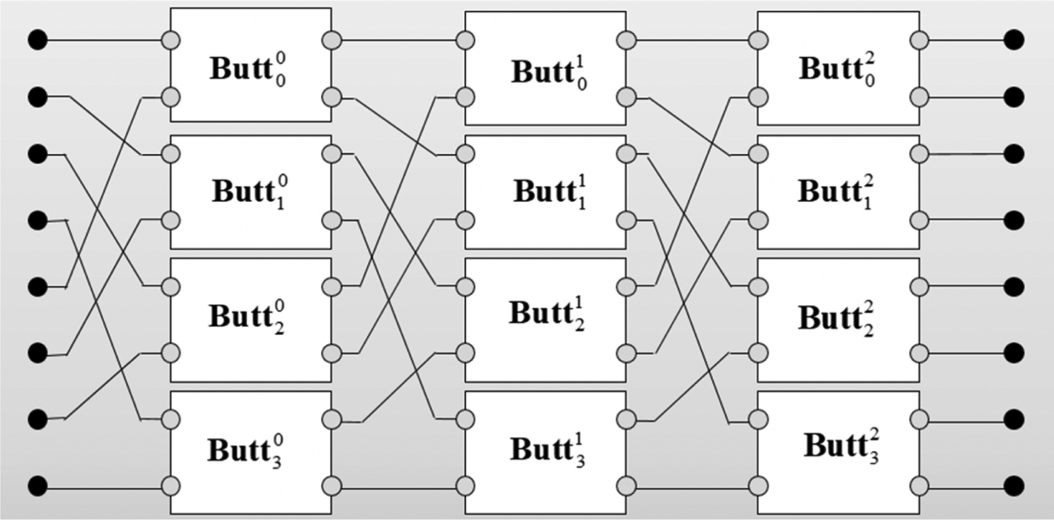


Рис. 2. 8-точечное быстрое преобразование Фурье (с взаимосвязями между бабочками)

Очевидно, что как *прямое, так и обратное преобразование* $\mathrm{BT}_2(h,f)$включают *только* две *произвольные* функции $f,h$, а *не* их обратные.

Для применения нелинейного тензорного произведение суперпозиции определяются два элементарных нелинейных $(2\times 2)$ -преобразования, называемых нелинейными*преобразованиями-бабочками*:$\mathrm{BT}_1 : \mathrm{V}_2 \to \mathrm{V}_2$, $\mathrm{BT}_2 : \mathrm{V}_2 \to \mathrm{V}_2$. Каждое из них описывается двумя *произвольными* функциями $\mathrm{BT}_2^1\left[g_0^1,g_1^1\right]$, $\mathrm{BT}_2^2\left[g_0^2,g_1^2\right]$. Обозначим их матрично:

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \mathrm{BT}_2^1\left[g_0^1,g_1^1\right]\circ\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} =$$

$$= \begin{bmatrix} {}_{0\leftarrow} g_0^1(\cdot_0,\ \cdot_1) \\ {}_{1\leftarrow} g_1^1(\cdot_0,\ \cdot_1) \end{bmatrix}\circ\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} {}_{0\leftarrow} g_0^1(x_0,x_1) \\ {}_{1\leftarrow} g_1^1(x_0,x_1) \end{bmatrix},$$

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \mathrm{BT}_2^2\left[g_0^2,g_1^2\right]\circ\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} =$$

$$= \begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix}\circ\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} {}_{0\leftarrow} g_0^2(x_0,x_1) \\ {}_{1\leftarrow} g_1^1(x_0,x_1) \end{bmatrix}.$$

Иногда авторы будут использовать веса

$$w_{00}^1 = -w_{11}^1 = \cos\theta^1, w_{01}^1 = w_{10}^1 = \sin\theta^1;$$

$$\text{и } w_{00}^2 = -w_{11}^2 = \cos\theta^2, w_{01}^2 = w_{10}^2 = \sin\theta^2,$$

$$w_{01}^2 = w_{10}^2 = \sin\theta^2, \text{ для переменных } x_0,x_1$$

и «амплитуды» $a_0^1 = e^{i\varphi_0^1}$, $a_1^1 = e^{i\varphi_1^1}$, $a_0^2 = e^{i\varphi_0^2}$, $a_1^2 = e^{i\varphi_1^2}$ для нелинейных функций:

$$\begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} {}_{0\leftarrow} a_0^k\cdot g_0^k\left(w_{00}^k(\cdot_0),w_{01}^k(\cdot_1)\right) \\ {}_{1\leftarrow} a_1^k\cdot g_1^k\left(w_{10}^k(\cdot_0),w_{11}^k(\cdot_1)\right) \end{bmatrix}\circ\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} =$$

$$= \begin{bmatrix} {}_{0\leftarrow} a_0^k\cdot g_0^k\left(w_{00}^k\cdot x_0,w_{01}^k\cdot x\right) \\ {}_{1\leftarrow} a_1^k\cdot g_1^k(w_{10}^k\cdot x_0,w_{11}^k\cdot x_1) \end{bmatrix},$$

где $k = 1,2$. В данном случае $\mathrm{BT}_2^1\left[\varphi_0^1,\varphi_1^1,\theta^1;\ g_0^1,g_1^1\right]$ и $\mathrm{BT}_2^2\left[\varphi_0^2,\varphi_1^2,\theta^2;\ g_0^2,g_1^2\right]$ зависят от 3 скалярных параметров $\theta = (\varphi_0,\varphi_1,\theta)\in[0,2\pi]^3$ и двух нелинейных функций $\mathbf{g} = (g_0,g_1)$.

**Определение 1.** Тензорное произведение двух нелинейных $(2\times 2)$ -преобразований $\mathrm{BT}_2^1\left[g_0^1,g_1^1\right]$ и $\mathrm{BT}_2^2\left[g_0^2,g_1^2\right]$ является нелинейным $(4\times 4)$ -преобразованием, которое определяется как

$$\mathrm{BT}_2^1\left[g_0^1,g_1^1\right]\otimes\mathrm{BT}_2^2\left[g_0^2,g_1^2\right] =$$

$$= \begin{bmatrix} {}_{0\leftarrow} g_0^1(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^1(\cdot_0,\ \cdot_1) \end{bmatrix}\otimes\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix} :=$$

$$= \begin{bmatrix} {}_{0\leftarrow} g_0^1\left(\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix}_0,\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix}_1\right) \\ \hdashline {}_{1\leftarrow} g_1^1\left(\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix}_0,\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_0,\ \cdot_1) \end{bmatrix}_1\right) \end{bmatrix} =$$

$$= \begin{bmatrix} {}_{0\leftarrow} g_0^1\left(\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_{00},\ \cdot_{01}) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_{00},\ \cdot_{01}) \end{bmatrix},\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_{10},\ \cdot_{11}) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_{10},\ \cdot_{11}) \end{bmatrix}\right) \\ \hdashline {}_{1\leftarrow} g_1^1\left(\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_{00},\ \cdot_{01}) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_{00},\ \cdot_{01}) \end{bmatrix},\begin{bmatrix} {}_{0\leftarrow} g_0^2(\cdot_{10},\ \cdot_{11}) \\ \hdashline {}_{1\leftarrow} g_1^2(\cdot_{10},\ \cdot_{11}) \end{bmatrix}_1\right) \end{bmatrix} =$$

$$= \begin{bmatrix} {}_{00\leftarrow} g_0^1\left(g_0^2(\cdot_{00},\ \cdot_{01}),g_0^2(\cdot_{10},\ \cdot_{11})\right) \\ \hdashline {}_{01\leftarrow} g_0^1\left(g_1^2(\cdot_{00},\ \cdot_{01}),g_1^2(\cdot_{10},\ \cdot_{11})\right) \\ \hdashline {}_{10\leftarrow} g_1^1\left(g_0^2(\cdot_{00},\ \cdot_{01}),g_0^2(\cdot_{10},\ \cdot_{11})\right) \\ \hdashline {}_{11\leftarrow} g_1^1\left(g_1^2(\cdot_{00},\ \cdot_{01}),g_1^2(\cdot_{10},\ \cdot_{11})\right) \end{bmatrix},$$

где

$$g_i^1\left(\left[\begin{array}{c} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_0,\left[\begin{array}{c} g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_1\right)=$$

$$= g_i^1\left(\left[\begin{array}{c} g_0^2(\cdot_{00},\ \cdot_{01}) \\ \hdashline g_1^2(\cdot_{00},\ \cdot_{01})\end{array}\right],\left[\begin{array}{c} g_0^2(\cdot_{10},\ \cdot_{11}) \\ \hdashline g_1^2(\cdot_{10},\ \cdot_{11})\end{array}\right]\right):=$$

$$:=\begin{bmatrix} g_i^1\left(g_0^2(\cdot_{00},\ \cdot_{01}),g_0^2(\cdot_{10},\ \cdot_{11})\right) \\ g_i^1\left(g_1^2(\cdot_{00},\ \cdot_{01}),g_1^2(\cdot_{10},\ \cdot_{11})\right)\end{bmatrix}=$$

$$=\begin{bmatrix} g_i^1\left(g_0^2,g_0^2\right) \\ g_i^1\left(g_1^2,g_1^2\right)\end{bmatrix}=\begin{bmatrix} g_i^1 \overset{\leftarrow}{\bullet} g_0^2 \\ g_i^1 \overset{\leftarrow}{\bullet} g_1^2\end{bmatrix}$$

— правило композиции адресов входных данных и

$$\left[\begin{array}{c} {}_{0\leftarrow}g_0^1\left(\left[\begin{array}{c} {}_{0\leftarrow}g_0^2 \\ \hdashline {}_{1\leftarrow}g_1^2\end{array}\right],\left[\begin{array}{c} {}_{0\leftarrow}g_0^2 \\ \hdashline {}_{1\leftarrow}g_1^2\end{array}\right]\right) \\ \hline {}_{1\leftarrow}g_1^1\left(\left[\begin{array}{c} {}_{0\leftarrow}g_0^2 \\ \hdashline {}_{1\leftarrow}g_1^2\end{array}\right],\left[\begin{array}{c} {}_{0\leftarrow}g_0^2 \\ \hdashline {}_{1\leftarrow}g_1^2\end{array}\right]\right)\end{array}\right]:=$$

$$=\left[\begin{array}{c} {}_{00\leftarrow}g_0^1\left(g_0^2,g_0^2\right) \\ \hdashline {}_{01\leftarrow}g_0^1\left(g_1^2,g_1^2\right) \\ \hline {}_{10\leftarrow}g_1^1\left(g_0^2,g_0^2\right) \\ \hdashline {}_{11\leftarrow}g_1^1\left(g_1^2,g_1^2\right)\end{array}\right]=\left[\begin{array}{c} {}_{00\leftarrow}g_0^1 \overset{\leftarrow}{\bullet} g_0^2 \\ \hdashline {}_{01\leftarrow}g_0^1 \overset{\leftarrow}{\bullet} g_1^2 \\ \hline {}_{10\leftarrow}g_1^1 \overset{\leftarrow}{\bullet} g_0^2 \\ \hdashline {}_{11\leftarrow}g_1^1 \overset{\leftarrow}{\bullet} g_1^2\end{array}\right]$$

— правило составления адресов выходных данных. Здесь по определению

$$g_0^1 \overset{\leftarrow}{\bullet} g_0^2 := g_0^1\left(g_0^2,g_0^2\right),\ g_0^1 \overset{\leftarrow}{\bullet} g_1^2 :=$$
$$= g_0^1\left(g_1^2,g_1^2\right),\ g_1^1 \overset{\leftarrow}{\bullet} g_0^2 := g_1^1\left(g_0^2,g_0^2\right),$$

$g_1^1 \overset{\leftarrow}{\bullet} g_1^2 := g_1^1\left(g_1^2,g_1^2\right)$ и $\overset{\leftarrow}{\bullet}$ является символом левой суперпозиции нелинейных функций. Так,

$$\mathbf{BT}_2^1\left[g_0^1,g_1^1\right] \overset{\leftarrow}{\otimes} \mathbf{BT}_2^2\left[g_0^2,g_1^2\right] = \mathbf{BT}_4\left[(g_0^1,g_1^1) \overset{\leftarrow}{\bullet} (g_0^2,g_1^2)\right]=$$
$$= \mathbf{BT}_4\left[g_0^1 \overset{\leftarrow}{\bullet} g_0^2,\ g_0^1 \overset{\leftarrow}{\bullet} g_1^2,\ g_1^1 \overset{\leftarrow}{\bullet} g_0^2,\ g_1^1 \overset{\leftarrow}{\bullet} g_1^2\right],$$

$$\mathbf{BT}_2^2\left[g_0^2,g_1^2\right] \overset{\leftarrow}{\otimes} \mathbf{BT}_2^1\left[g_0^1,g_1^1\right] = \mathbf{BT}_4\left[(g_0^2,g_1^2) \overset{\leftarrow}{\bullet} (g_0^1,g_1^1)\right]=$$
$$= \mathbf{BT}_4\left[g_0^2 \overset{\leftarrow}{\bullet} g_0^1,\ g_0^2 \overset{\leftarrow}{\bullet} g_1^1,\ g_1^2 \overset{\leftarrow}{\bullet} g_0^1,\ g_1^2 \overset{\leftarrow}{\bullet} g_1^1\right].$$

**Определение 2.** Тензорное произведение двух нелинейных преобразований $\mathbf{BT}_{N_1}^1\left[g_0^1,g_1^1,...,g_{N_1-1}^1\right]$ и $\mathbf{BT}_{N_2}^2\left[g_0^2,g_1^2,...,g_{N_2-1}^1\right]$ является нелинейным $(N_1 \times N_2)$-преобразованием, которое определяется как

$$\mathbf{BT}_{N_1}^1\left[g_0^1,g_1^1,...,g_{N_1-1}^1\right] \overset{\leftarrow}{\otimes} \mathbf{BT}_{N_2}^2\left[g_0^2,g_1^2,...,g_{N_2-1}^2\right]=$$
$$= \mathbf{BT}_{N_1N_2}\left[(g_0^1,g_1^1,...,g_{N_1-1}^1) \overset{\leftarrow}{\bullet} (g_0^2,g_1^2,...,g_{N_2-1}^2)\right]= \quad (5)$$
$$= \mathbf{BT}_{N_1N_2}\left[\left\{g_i^1 \overset{\leftarrow}{\bullet} g_j^2\right\}_{i=0,\quad j=0}^{N_1-1,\ N_2-1}\right],$$

где $g_i^1 \overset{\leftarrow}{\bullet} g_j^2 = g_i^1(\cdot_0,\cdot_1,...,\cdot_{N_1-1}) \overset{\leftarrow}{\bullet} g_j^2 := g_i^1\left(g_j^2,g_j^2,...,g_j^2\right)$.

**Определение 3.** Определим $n$-кратное тензорное произведение $n$ нелинейных преобразований $\left\{\mathbf{BT}_{N_p}^p\left[g_0^p,g_1^p,...,g_{N_p-1}^p\right]\right\}_{p=1}^n$ как следующее нелинейное $(N \times N)$-преобразование

$$\mathbf{BT}_{N_1}^1\left[g_0^1,g_1^1,...,g_{N_1-1}^1\right] \overset{\leftarrow}{\otimes} ... \overset{\leftarrow}{\otimes} \mathbf{BT}_{N_n}^n\left[g_0^n,g_1^n,...,g_{N_2-1}^n\right]=$$
$$= \mathbf{BT}_{N_1N_2...N_n}\left[\left\{g_{i_1}^1 \overset{\leftarrow}{\bullet} g_{i_2}^2 \overset{\leftarrow}{\bullet} ... \overset{\leftarrow}{\bullet} g_{i_n}^n\right\}_{i_1=0,\quad i_2=0,\ ...,\ i_n=0}^{N_1-1,\ N_2-1,\ ...,\ N_n-1}\right],$$

где $N = N_1N_2...N_n$.

*Пример 1.* Пусть $\mathbf{BT}_2^1\left[g_0^1,g_1^1\right]$ — тождественное преобразование

$$\mathbf{BT}_2^1\left[g_0^1,g_1^1\right] \equiv \mathbf{I}_{2\times 2} = \begin{bmatrix} {}_{0\leftarrow}g_0^1(\cdot_0,\ \cdot_1) \\ {}_{1\leftarrow}g_1^1(\cdot_0,\ \cdot_1)\end{bmatrix}=$$
$$=\begin{bmatrix} {}_{0\leftarrow}\mathbf{Add}_0\left(1(\cdot_0),0(\cdot_1)\right) \\ {}_{1\leftarrow}\mathbf{Add}_1\left(0(\cdot_0),1(\cdot_1)\right)\end{bmatrix},$$

затем

$$\mathbf{I}_{2\times 2} \overset{\leftarrow}{\otimes} \mathbf{BT}_2^2\left[g_0^2,g_1^2\right]=$$
$$=\left[\begin{array}{c} {}_{0\leftarrow}\mathbf{Add}_0\left(1(\cdot_0),0(\cdot_1)\right) \\ \hdashline {}_{1\leftarrow}\mathbf{Add}_1\left(0(\cdot_0),1(\cdot_1)\right)\end{array}\right] \overset{\leftarrow}{\otimes} \left[\begin{array}{c} {}_{0\leftarrow}g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]=$$

$$=\left[\begin{array}{c} {}_{0\leftarrow}\mathbf{Add}_0\left(1\cdot\left[\begin{array}{c} {}_{0\leftarrow}g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_0,0\cdot\left[\begin{array}{c} {}_{0\leftarrow}g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_1\right) \\ \hdashline {}_{1\leftarrow}\mathbf{Add}_1^1\left(0\cdot\left[\begin{array}{c} {}_{0\leftarrow}g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_0,1\cdot\left[\begin{array}{c} {}_{0\leftarrow}g_0^2(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^2(\cdot_0,\ \cdot_1)\end{array}\right]_1\right)\end{array}\right]=$$

$$=\left[\begin{array}{c} {}_{00\leftarrow}g_0^2(\cdot_{00},\ \cdot_{01}) \\ \hdashline {}_{01\leftarrow}g_1^2(\cdot_{00},\ \cdot_{01}) \\ \hdashline {}_{10\leftarrow}g_0^2(\cdot_{10},\ \cdot_{11}) \\ \hdashline {}_{11\leftarrow}g_1^2(\cdot_{10},\ \cdot_{11})\end{array}\right].$$

Если $\mathbf{BT}_2^2\left[g_0^2,g_1^2\right]$ — идентичное преобразование, то

$$\mathbf{BT}_2^1\left[g_0^1,g_1^1\right] \overset{\leftarrow}{\otimes} \mathbf{I}_{2\times 2}=$$
$$=\left[\begin{array}{c} {}_{0\leftarrow}g_0^1(\cdot_0,\ \cdot_1) \\ \hdashline {}_{1\leftarrow}g_1^1(\cdot_0,\ \cdot_1)\end{array}\right] \otimes \left[\begin{array}{c} {}_{0\leftarrow}\mathbf{Add}_0\left(1(\cdot_0),0(\cdot_1)\right) \\ \hdashline {}_{1\leftarrow}\mathbf{Add}_1\left(0(\cdot_0),1(\cdot_1)\right)\end{array}\right]=$$

$$=\left[\begin{array}{c} {}_{00\leftarrow}g_0^1\left(\mathbf{Add}_0\left(1(\cdot_{00}),0(\cdot_{01})\right),\mathbf{Add}_0\left(1(\cdot_{10}),0(\cdot_{11})\right)\right) \\ \hdashline {}_{01\leftarrow}g_0^1\left(\mathbf{Add}_1\left(0(\cdot_{00}),1(\cdot_{01})\right),\mathbf{Add}_1\left(0(\cdot_{10}),1(\cdot_{11})\right)\right) \\ \hdashline {}_{10\leftarrow}g_1^1\left(\mathbf{Add}_0\left(1(\cdot_{00}),0(\cdot_{01})\right),\mathbf{Add}_0\left(1(\cdot_{10}),0(\cdot_{11})\right)\right) \\ \hdashline {}_{11\leftarrow}g_1^1\left(\mathbf{Add}_1\left(0(\cdot_{00}),1(\cdot_{01})\right),\mathbf{Add}_1\left(0(\cdot_{10}),1(\cdot_{11})\right)\right)\end{array}\right]=$$

$$=\left[\begin{array}{c} {}_{00\leftarrow}g_0^1\left(\cdot_{00},\cdot_{10}\right) \\ \hdashline {}_{01\leftarrow}g_0^1\left(\cdot_{01},\cdot_{11}\right) \\ \hdashline {}_{10\leftarrow}g_1^1\left(_{00},\cdot_{10}\right) \\ \hdashline {}_{11\leftarrow}g_1^1\left(\cdot_{01},\cdot_{11}\right)\end{array}\right].$$

Это легко проверить

$$\left(\mathrm{BT}_2^1\left[g_0^1, g_1^1\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{2\times 2}\right) \circ \left(\mathrm{I}_{2\times 2} \overset{\leftarrow}{\otimes} \mathrm{BT}_2^2\left[g_0^2, g_1^2\right]\right) =$$
$$= \mathrm{BT}_2^1\left[g_0^1, g_1^1\right] \overset{\leftarrow}{\otimes} \mathrm{BT}_2^2\left[g_0^2, g_1^2\right],$$

где $\mathrm{BT}_2^1\left[g_0^1, g_1^1\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{2\times 2}$ и — нелинейное разреженное $\mathrm{I}_{2\times 2} \overset{\leftarrow}{\otimes} \mathrm{BT}_2^2\left[g_0^2, g_1^2\right]$ *нелинейное преобразование по* основанию 2 ( **SNLT** ).

**Теорема 1.** Каждое *n*-кратное тензорное произведение $n$ нелинейных преобразований $\left\{\mathrm{BT}_{N_p}^p\left[g_0^p, g_1^p, ..., g_{N_p-1}^p\right]\right\}_{p=1}^n$ является $\circ$ –произведением разреженных нелинейных преобразований **SNLT**

$$
\begin{aligned}
&{}^{\mathrm{I}}\mathrm{FNLT}_{N_1N_2...N_n}\left[\left\{g_{i_1}^1 \overset{\leftarrow}{\bullet} g_{i_2}^2 \overset{\leftarrow}{\bullet} ... \overset{\leftarrow}{\bullet} g_{i_n}^n\right\}_{i_1=0,\ i_2=0,\ ...,\ i_n=0}^{N_1-1,\ N_2-1,\ ...,\ N_n-1}\right] = \\
&= \mathrm{BT}_{N_1}^1\left[g_0^1, g_1^1, ..., g_{N_1-1}^1\right] \overset{\leftarrow}{\otimes} ... \overset{\leftarrow}{\otimes} \mathrm{BT}_{N_n}^n\left[g_0^n, g_1^n, ..., g_{N_n-1}^n\right] = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ \left(\mathrm{I}_{\bar N_r \times \bar N_r} \overset{\leftarrow}{\otimes} \mathrm{BT}_{N_r}^r\left[g_0^r, ..., g_{N_r-1}^r\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{(\bar N_n/\bar N_r)\times(\bar N_n/\bar N_r)}\right) = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ\, \mathrm{SNLT}_{N_r}^r\left[g_0^r, ..., g_{N_r-1}^r\right] = \\
&= \mathrm{SNLT}_{N_n}^n\left[g_0^n, ..., g_{N_n-1}^n\right] \circ ... \\
&... \circ \mathrm{SNLT}_{N_2}^2\left[g_0^2, ..., g_{N_2-1}^2\right] \circ \mathrm{SNLT}_{N_1}^1\left[g_0^1, ..., g_{N_1-1}^1\right],
\end{aligned}
\tag{6}
$$

где

$$\mathrm{SNLT}_{N_r}^r\left[g_0^r, g_1^r, ..., g_{N_r-1}^r\right] =$$
$$= \mathrm{I}_{\bar N_r \times \bar N_r} \overset{\leftarrow}{\otimes} \mathrm{BT}_{N_r}^r\left[g_0^r, g_1^r, ..., g_{N_r-1}^r\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{(\bar N_n/\bar N_r)\times(\bar N_n/\bar N_r)}$$

это **FNLT**, где $\bar N_0 := 1$, $\bar N_{r-1} := N_1 N_2 ... N_{r-1}$, $r = 1, 2, ..., n$ и $\mathrm{I}_{1\times 1} \equiv 1$. В частности,

$$
\begin{aligned}
&{}^{\mathrm{I}}\mathrm{FNLT}_{2^n}\left[\left\{g_{i_1}^1 \overset{\leftarrow}{\bullet} g_{i_2}^2 \overset{\leftarrow}{\bullet} ... \overset{\leftarrow}{\bullet} g_{i_n}^n\right\}_{i_1=0,\ i_2=0, ..., i_n=0}^{i_1=1,\ i_2=1, ..., i_n=1}\right] = \\
&= \mathrm{BT}_2^1\left[g_0^1, g_1^1\right] \overset{\leftarrow}{\otimes} \mathrm{BT}_2^2\left[g_0^2, g_1^2\right] \overset{\leftarrow}{\otimes} ... \overset{\leftarrow}{\otimes} \mathrm{BT}_2^n\left[g_0^n, g_1^n\right] = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ \left(\mathrm{I}_{2^{r-1}\times 2^{r-1}} \overset{\leftarrow}{\otimes} \mathrm{BT}_2^r\left[g_0^r, g_1^r\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{2^{n-r}\times 2^{n-r}}\right) = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ\, \mathrm{SNLT}_2^r\left[g_0^r, g_1^r\right] = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ \left(\mathrm{I}_{2^{r-1}\times 2^{r-1}} \overset{\leftarrow}{\otimes} \mathrm{BT}_2^r\left[g_0^r, g_1^r\right] \overset{\leftarrow}{\otimes} \mathrm{I}_{2^{n-r}\times 2^{n-r}}\right) = \\
&= \prod_{r=1}^{n\leftarrow 1} \circ\, \mathrm{SNLT}_2^r\left[g_0^r, g_1^r\right] = \\
&= \mathrm{SNLT}_2^n\left[g_0^n, g_1^n\right] \circ \mathrm{SNLT}_2^2\left[g_0^2, g_1^2\right] \circ ... \circ \mathrm{SNLT}_2^1\left[g_0^1, g_1^1\right]
\end{aligned}
\tag{7}
$$

— это FNLT по основанию-2. Авторы определяют операцию «стержень» для $\mathrm{NLT}$ получения преобразования «нелинейного транспонирования».

$$\overline{\mathrm{NLT}}_{N_1N_2...N_n}\left[\left\{g_{i_1}^1 \overset{\leftarrow}{\bullet} g_{i_2}^2 \overset{\leftarrow}{\bullet} ... \overset{\leftarrow}{\bullet} g_{i_n}^n\right\}_{i_1=0,\ i_2=0,\ ...,\ i_n=0}^{N_1-1,\ N_2-1,\ ...,\ N_n-1}\right] =$$
$$= \mathrm{SNLT}_{N_r}^1\left[g_0^1, g_1^1, ..., g_{N_1-1}^1\right] \circ \mathrm{SNLT}_{N_2}^2\left[g_0^2, g_1^2, ..., g_{N_2-1}^2\right] \circ ...$$
$$... \circ \mathrm{SNLT}_{N_n}^n\left[g_0^n, g_1^n, ..., g_{N_n-1}^n\right] = \prod_{r=1}^{1\to n} \circ\, \mathrm{SNLT}_{N_r}^r\left[g_0^r, g_1^r, ..., g_{N_r-1}^r\right].$$

Пусть $\mathrm{V}_N = \mathrm{Span}\left\{e_0, e_1, ..., e_{N-1}\right\}$— $N$ пространство -D, натянутое на базисе $\left\{e_0, e_1, ..., e_{N-1}\right\}$. Авторы предполагают, что $N = 2^n$. Авторы делят пространство $\mathrm{V}^N$ на $M = 2^{n-1}$ двумерные пространства.

$$\mathrm{V}_N = \mathrm{V}_2^0\left\{e_{i_0^0}, e_{i_1^0}\right\} \oplus \mathrm{V}_2^1\left\{e_{i_0^1}, e_{i_1^1}\right\} \oplus ... \oplus \mathrm{V}_2^{M-1}\left\{e_{i_0^{M-1}}, e_{i_1^{M-1}}\right\}.$$

Далее введем следующие базисные $(2\times 2)$ преобразования:

$$\mathrm{BT}_2 = \left[j_0, j_1 \middle| \mathrm{BT}_2 \middle| i_0, i_1\right] =$$
$$= \left[j_0, j_1 \middle| g_0, g_1 \middle| i_0, i_1\right] : \mathrm{V}_2\{e_{i_0}, e_{i_1}\} \to \mathrm{V}_2\{e_{j_0}, e_{j_1}\},$$

действуя как

$$\begin{bmatrix} y_{j_0} \\ y_{j_1} \end{bmatrix} = \mathrm{BT}_2 \circ \begin{bmatrix} x_{i_0} \\ x_{i_1} \end{bmatrix} =$$
$$= \left[j_0, j_1 \middle| g_0, g_1 \middle| i_0, i_1\right] \circ \begin{bmatrix} x_{i_0} \\ x_{i_1} \end{bmatrix} = \begin{bmatrix} j_0 \leftarrow g(x_{i_0}, x_{i_1}) \\ j_1 \leftarrow g(x_{i_0}, x_{i_1}) \end{bmatrix}.$$

Семейство $\mathcal{F} = \left\{\left[j_0^p, j_1^p \middle| g_0^p, g_1^p \middle| i_0^p, i_1^p\right]\right\}_{p=0}^S$ произвольных $S = M - 1 = 2^{n-1} - 1$ базисных преобразований генерируется в $\mathrm{V}_N$ некотором нелинейном **SNLT**.

$$\mathrm{SNLT}_{2^n} = \left[\begin{array}{c} \left[j_0^0, j_1^0 \middle| g_0^0, g_1^0 \middle| i_0^0, i_1^0\right] \\ \hdashline \left[j_0^1, j_1^1 \middle| g_0^1, g_1^1 \middle| i_0^1, i_1^1\right] \\ \hdashline \vdots \\ \hdashline \left[j_0^S, j_1^S \middle| g_0^S, g_1^S \middle| i_0^S, i_1^S\right] \end{array}\right] =$$
$$= \bigoplus_{p=0}^S \mathrm{BT}_2^p = \bigoplus_{p=0}^S \left[j_0^p, j_1^p \middle| g_0^p, g_1^p \middle| i_0^p, i_1^p\right].$$

Каждое разреженное преобразование содержит $S$ базисные преобразования. Позволять

$$\mathrm{BT}_{2^n} := \left[j_0^p, j_1^p \middle| \mathrm{BT}_{2^n} \middle| i_0^p, i_1^p\right] := \mathrm{I}_{2p} \oplus \mathrm{BT}_2 \oplus \mathrm{I}_{2^n-2(p+1)} =$$
$$= \mathrm{I}_{2p} \oplus \left[j_0^p, j_1^p \middle| \mathrm{BT}_2 \middle| i_0^p, i_1^p\right] \oplus \mathrm{I}_{2^n-2(p+1)} =$$
$$= \mathrm{I}_{2p} \oplus \left[j_0^p, j_1^p \middle| g_0^p, g_1^p \middle| i_0^p, i_1^p\right] \oplus \mathrm{I}_{2^n-2(p+1)}.$$

Тогда разреженные матрицы можно представить как $\circ$ –произведение суперпозиции.

$$\mathrm{SNLT}_{2^n} = \prod_{p=0}^S \circ \mathrm{BT}_{2^n}^p =$$
$$= \prod_{p=0}^S \circ \left(\mathrm{I}_{2p} \oplus \left[j_0^p, j_1^p \middle| \mathrm{BT}_2^p \middle| i_0^p, i_1^p\right] \oplus \mathrm{I}_{2^n-2(p+1)}\right) =$$
$$= \prod_{p=0}^S \circ \left(\mathrm{I}_{2p} \oplus \left[j_0^p, j_1^p \middle| g_0^p, g_1^p \middle| i_0^p, i_1^p\right] \oplus \mathrm{I}_{2^n-2(p+1)}\right).$$

Две последовательности $\left\{\begin{matrix} i_0^0,\ i_1^0;\ i_0^1,\ i_1^1;\ldots;i_0^S,\ i_1^S \\ j_0^0,\ j_1^0;\ j_0^1,\ j_1^1;\ldots;j_0^S,\ j_1^S \end{matrix}\right\}$ образуют две схемы адресации базисных преобразований $\left\{\mathrm{BT}_2^p\right\}_{p=0}^S$ для так называемого быстрого **NLT по основанию 2 без места** .

**Определение 4.** Пусть

$$\left\{\left\{ {}^r\mathrm{BT}_2^p = \left[ {}^r j_0^p, {}^r j_1^p \middle| {}^r g_0^p, {}^r g_1^p \middle| {}^r i_0^p, {}^r i_1^p \right] \right\}_{p=0}^S \right\}_{r=1}^L$$

— множество $L$ семейств произвольных базисных преобразований. Определим обобщенное тензорное произведение этих семейств в виде суперпозиции $\circ$-произведения разреженных нелинейных преобразований:

$$ {}^{\mathrm{II}}\mathrm{FNLT}_{2^n} = \prod_{r=1}^{L}\circ\prod_{p=0}^{S}\circ\, {}^r\mathrm{BT}_{2^n}^p = \prod_{r=1}^{L}\circ\, {}^r\mathrm{SNLT}_{2^n} = \qquad (8)$$

$$= \prod_{r=1}^{L}\circ\prod_{p=0}^{S}\circ\left(\mathrm{I}_{2p} \oplus \left[ {}^r j_0^p, {}^r j_1^p \middle| {}^r\mathrm{BT}_2^p \middle| {}^r i_0^p, {}^r i_1^p \right] \oplus \mathrm{I}_{2^n-2(p+1)}\right).$$

Это называется быстрыми нелинейными преобразованиями по основанию-2, $L$ которые ($L=\log_2 N = n$ обычно используются для линейных унитарных преобразований). В деталях это быстрое преобразование имеет следующий вид:

$$ {}^{\mathrm{II}}\mathrm{FNLT} = \prod_{r=1}^{L\leftarrow 1}\circ\, {}^r\mathrm{SNLT} =$$

$$= {}^L\mathrm{SNLT}_{2^n} \circ \ldots \circ {}^2\mathrm{SNLT}_{2^n} \circ {}^1\mathrm{SNLT}_{2^n} =$$

$$= \begin{array}{|c|}\hline {}^L\mathrm{BT}_2^0 \\ \hline {}^L\mathrm{BT}_2^1 \\ \hline \vdots \\ \hline {}^L\mathrm{BT}_2^S \\ \hline \end{array} \circ \ldots \circ \begin{array}{|c|}\hline {}^2\mathrm{BT}_2^0 \\ \hline {}^2\mathrm{BT}_2^1 \\ \hline \vdots \\ \hline {}^2\mathrm{BT}_2^S \\ \hline \end{array} \circ \begin{array}{|c|}\hline {}^1\mathrm{BT}_2^0 \\ \hline {}^1\mathrm{BT}_2^1 \\ \hline \vdots \\ \hline {}^1\mathrm{BT}_2^S \\ \hline \end{array} =$$

$$= \prod_{r=1}^{L\leftarrow 1}\circ \begin{array}{|c|}\hline \left[ {}^r j_0^0, {}^r j_1^0 \middle| {}^r g_0^0, {}^r g_1^0 \middle| {}^r i_0^0, {}^r i_1^0 \right] \\ \hline \left[ {}^r j_0^1, {}^r j_1^1 \middle| {}^r g_0^1, {}^r g_1^1 \middle| {}^r i_0^1, {}^r i_1^1 \right] \\ \hline \vdots \\ \hline \left[ {}^r j_0^S, {}^r j_1^S \middle| {}^r g_0^S, {}^r g_1^S \middle| {}^r i_0^S, {}^r i_1^S \right] \\ \hline \end{array}. \qquad (9)$$

При линейной цифровой обработке сигналов используются следующие схемы адресации [4–8]

$$\begin{cases} {}^r j_0^p = {}^r i_0^p = p_r, \\ {}^r j_1^p = {}^r i_1^p = p_r + 2^{r-1}, \end{cases} \quad \begin{cases} {}^r j_0^p = 2p, & {}^r i_0^p = p_r, \\ {}^r j_1^p = 2p+1 & {}^r i_1^p = p_r + 2^{r-1}, \end{cases}$$

$$\begin{cases} {}^r j_0^p = p_r, & {}^r i_0^p = 2p, \\ {}^r j_1^p = p_r + 2^{r-1} & {}^r i_1^p = 2p+1. \end{cases}$$

где $p_r = 2^r \left\lfloor p / 2^{r-1} \right\rfloor + (p \bmod 2^{r-1})$. Для этих схем авторы предлагают следующую классическую архитектуру для быстрых нелинейных преобразований:

$$ {}^{(1)}\mathrm{FNLT}_{2^n} =$$

$$= \prod_{r=1}^{n}\prod_{p=0}^{S}\circ\left(\begin{matrix} \mathrm{I}_{2p} \oplus \left[ p_r, p_r + 2^{r-1} \middle| {}^r\mathrm{BT}_2^p \middle| p_r, p_r + 2^{r-1} \right] \oplus \\ \oplus\, \mathrm{I}_{2^n-2(p+1)} \end{matrix}\right),$$

$$ {}^{(2)}\mathrm{FNLT}_{2^n} =$$

$$= \prod_{r=1}^{n}\prod_{p=0}^{S}\circ\left(\mathrm{I}_{2p} \oplus \left[ {}^r\mathrm{BT}_2^p \middle| p_r, p_r + 2^{r-1} \right] \oplus \mathrm{I}_{2^n-2(p+1)}\right),$$

$$ {}^{(3)}\mathrm{FNLT}_{2^n} =$$

$$= \prod_{r=1}^{n}\prod_{p=0}^{S}\circ\left(\mathrm{I}_{2p} \oplus \left[ p_r, p_r + 2^{r-1} \middle| {}^r\mathrm{BT}_2^p \right] \oplus \mathrm{I}_{2^n-2(p+1)}\right),$$

где $\prod_{r=1}^{n} = \prod_{r=1}^{n\leftarrow 1}$ или $\prod_{r=1}^{n} = \prod_{r=1}^{1\to n}$. На рисунке 3 показан поток данных для системы счисления-2. ${}^{(1)}\mathrm{FNLT}_8$.

Широкое семейство обратимых базисных преобразований $\left[ j_0^p, j_1^p \middle| \mathrm{BT}_2^p \middle| i_0^p, i_1^p \right]$ можно найти в [2]. Используя их, $\mathrm{BT}_2^p$ авторы получают обратимые быстрые нелинейные преобразования $\mathrm{FNLT}_{2^n}$.

## Заключение

Создан новый подход к преобразованиям с нелинейными зависимостями и использованием быстрого алгоритма. Применяемые правила рекурсии для моделей нелинейных преобразований создают новые возможности для регулярного просмотра неявных быстрых преобразований с нелинейностью. Таким образом, авторы сформулировали новое свойство преобразований, определяемое как результат суперпозиции «разреженных» преобразований с нелинейностью, и создать новый (с меньшим количеством вычислительных операций) быстрый алгоритм NLWT. Авторы считают, что предложенные преобразования с нелинейностью, используемые для обработки в пространстве пикселей существенно упрощает процесс распознавания и обработки изображений, позволяет выполнять сложные операции с данными (нелинейная обработка), с высокой эффективностью.

Предлагаемая модель может применяться в компьютерных сетях, в технологиях искусственного интеллекта и как следствие повышает эффективность как проводной, так и беспроводной связи по которым передаются видеопотоки данных.

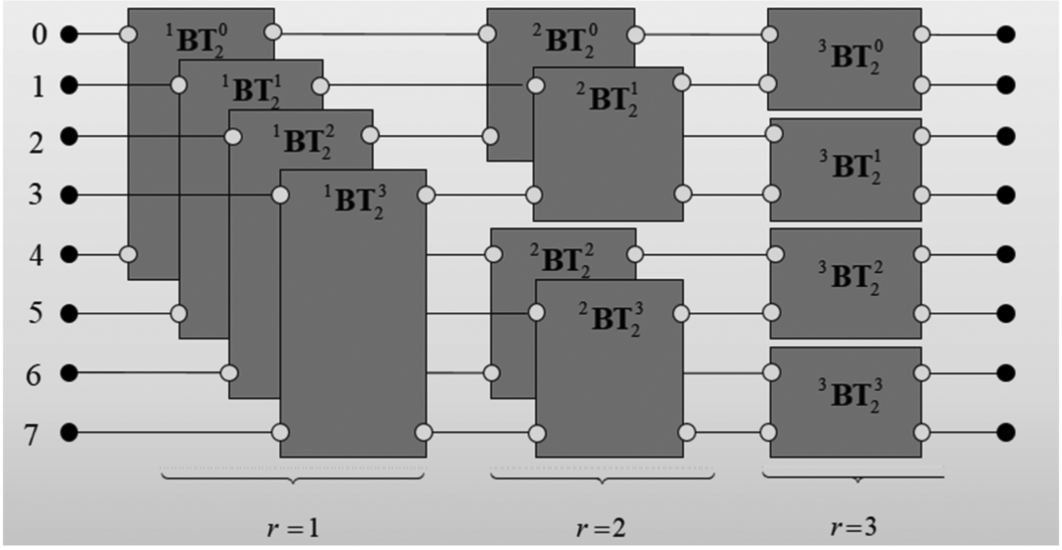


Рис. 3. Радикс-2 $^{(1)}\mathbf{FNLT}_8 = \prod_{r=1}^{1\to 3}\circ\prod_{p=0}^{3}\circ\left(\mathbf{I}_{2p}\oplus\left[{}^{r}\mathbf{BT}_2^{p}\right]\oplus\mathbf{I}_{8-2(p+1)}\right)$

ЛИТЕРАТУРА

1. Лабунец В.Г. Интеллектуальная технология OFDM, основанная на быстрых многопараметрических преобразованиях. — Екатеринбург: Издательство ИП Пиджаков А.В., 2023. 440 с.
2. Лабунец В.Г, Часовских В.П. Является ли мозг классическим компьютером, работающим в алгебре Клиффорда? Математические основы теории. — Екатеринбург: Издательство ООО «Акдениз», 2022. 178 с.
3. Лабунец В. Остеймер Э. Системный подход к нелинейной фильтрации, связанной с операторами агрегации. Часть 1. SISO-фильтры // Procedia Engineering. 2017. № 201. С.385–397.
4. Лабунец В. Остеймер Э. Системный подход к нелинейной фильтрации, связанной с операторами агрегации. Часть 2. MIMO-фильтры Фреше // Procedia Engineering. 2017. № 201. С.397–411.
5. Остеймер Э., Лабунец В., Комаров Д., Федорова Т. Фильтры Фреше для фильтрации цветных и гиперспектральных изображений // Коммуникации в компьютерных и информационных науках . 2015. № 542. С.57–70.
6. Остхаймер Э., Лабунец В., Мясников Ф. Семейства цифровых фильтров Герона для фильтрации изображений // Материалы семинара CEU. 2015. № 1452. С.56–63.
7. Лабунец В.Г. Гиперкомплексные модели многоканальных изображений // Труды института математики и механики УрО РАН. 2020. Т. 26. № 3. С.69–83.
8. Евдокимова А.Е., Зобова Л.Л. «Эффект бабочки» и возможность прогнозов в экономике // Современные наукоемкие технологии. 2014. № 7-3. С.72–73
9. Лабунец В.Г., Кох Е.В., Остхаймер Е. Алгебраические модели и методы компьютерной обработки изображений. Часть 1. Мультиплетные модели многоканальных изображений // Компьютерная оптика. 2018. Т. 42. № 1. С.84–95.

© Часовских Виктор Петрович (u2007u@ya.ru); Кох Елена Викторовна (elenakox@mail.ru)
Журнал «Современная наука: актуальные проблемы теории и практики»