

# ЛР4: Обработка изображений в частотной области.

Цель: Использование дискретного косинусного преобразования для обработки изображений.

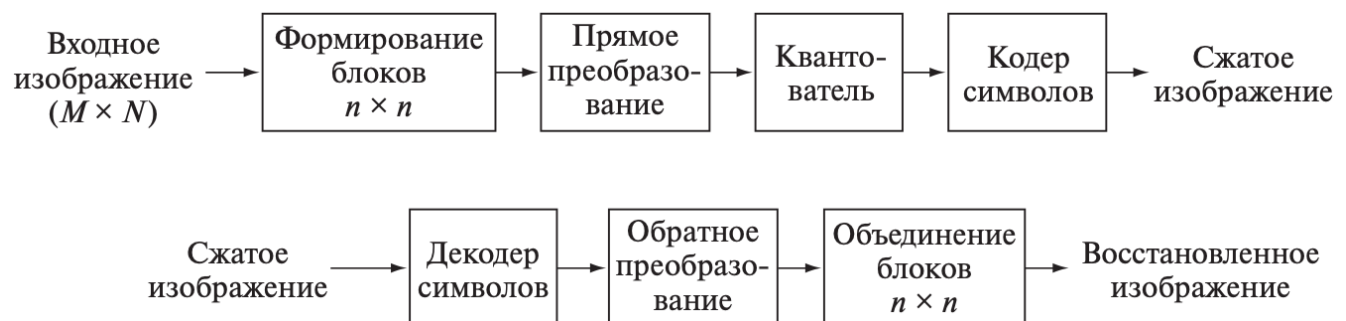
## Теоретическая часть

### Блочное трансформационное кодирование

Блочное трансформационное кодирование используется в следующих стандартах сжатия и файловых форматах: JPEG, М-JPEG, MPEG-1,2,4, H.264 и т.д. В данной работе будет рассмотрен метод сжатия изображений, используемый в стандарте JPEG. Метод сжатия основан на разбиении изображения на небольшие неперекрывающиеся блоки одинаковых размеров ( $8 \times 8$ ) и последующей независимой обработке блоков при помощи двумерных преобразований (ДКП).

В блочном трансформационном кодировании обратимое линейное преобразование (ДКП) используется для отображения каждого блока в набор коэффициентов преобразования, которые затем квантуются и кодируются. Для большинства реальных изображений значительное число коэффициентов имеют малую величину и могут быть достаточно грубо квантованы (или полностью удалены) ценой небольшого искажения изображения.

На рисунке схематически показана типичная система блочного трансформационного кодирования. Кодер выполняет четыре достаточно понятные операции: разбиение изображения на блоки, преобразование, квантование и кодирование. Декодер выполняет обратную последовательность операций (за исключением квантования). Первоначально изображение размерами  $M \times N$  разбивается на  $MN/n^2$  блоков размерами  $n \times n$ , которые затем и подвергаются преобразованиям.



Целью процесса преобразования является декорреляция значений элементов в каждом блоке или уплотнение как можно большего количества информации в наименьшее число коэффициентов преобразования. Изображение после преобразования и перетасовки будет состоять из 64 областей (8x8). Верхний левый угол содержит строковую и столбцовую НЧ часть изображения. Нижний правый угол содержит строковую и столбцовую ВЧ часть изображения.



На этапе квантования грубо квантуются или же удаляются те коэффициенты, которые несут малое количество информации в заранее заданном смысле (ВЧ часть изображения). Эти коэффициенты дают наименьший вклад в качество восстанавливаемого блока. На заключительном этапе осуществляется кодирование квантованных коэффициентов.

### Дискретное косинусное преобразование (ДКП)

Прямое дискретное косинусное преобразование может быть выражено в следующем виде:

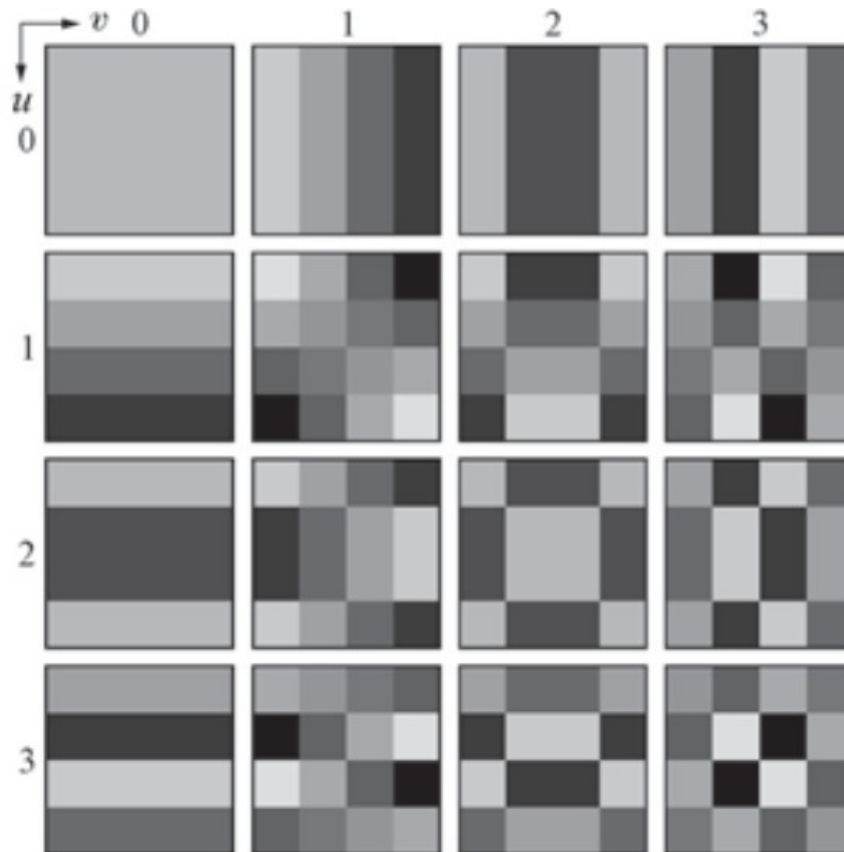
$$B_{pq} = \alpha(p, M)\alpha(q, N) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos \frac{\pi(2m+1)p}{2M} \cos \frac{\pi(2n+1)q}{2N}$$

$$\alpha(x, n) = \begin{cases} \sqrt{1/n}, & \text{if } x = 0, \\ \sqrt{2/n}, & \text{else } x = 1, 2, \dots, n-1 \end{cases}$$

$$(0 \leq p \leq M-1); (0 \leq q \leq N-1)$$

где  $A$  — входное изображение размера  $M \times N$ ;  $B$  — выходное изображение размера  $P \times Q$ ;  $m$  и  $n$  — координаты пикселя входного изображения;  $p$  и  $q$  — координаты пикселя выходного изображения.

Базисные функции дискретного косинусного преобразования для  $n = 4$  приведены на рисунке. Начало координат каждого блока находится в его левом верхнем углу.



### Ход работы

#### 1. Использование ДКП для анализа изображений:

- a) Загрузить все изображение (*lena\_gray\_256.tif*) с помощью команды *imread*.
- b) Разбить полученное изображение на блоки размерами  $8 \times 8$ .
- c) Для каждого блока выполнить ДКП с помощью команды *dct2*.
- d) Перетасовка пикселей:
  1. Пусть на входе имеется изображение  $256 \times 256$  пикселей, после вычисления ДКП.
  2. Рассчитаем количество блоков по горизонтали  $g=256/8=32$  и вертикали  $v=256/8=32$ .
  3. Перетасовать пиксели всех блоков, чтобы в результате получилось изображение состоящее из  $8 \times 8$  областей ( $256 \times 256$  пикселей). Каждая

область будет размерами 32x32 пикселя, состоящая из пикселей всех блоков. Например пиксель (1,1) всех блоков должны быть сосредоточены также в области (1,1).

4. Отобразить полученное изображение.

**2. Использование ДКП для синтеза изображений:**

- a) Выполнить операцию обратную перетасовке пикселей (анализ этап d). На входе имеем изображение состоящее из 8x8 областей (256x256 пикселей). После перетасовки получим изображение состоящее из 32x32 блоков.
- b) Для каждого блока выполнить ОДКП с помощью команды *idct2*.
- c) Преобразовать блоки в восстановленное изображение.
- d) Сравнить восстановленное и исходное изображения.

**3. Написать собственную реализацию ДКП, в соответствии с уравнением приведенным выше, и сравнить с реализацией в MATLAB.**