

# Преобразование Фурье. JPEG

Влад Шахуро



15 сентября 2017

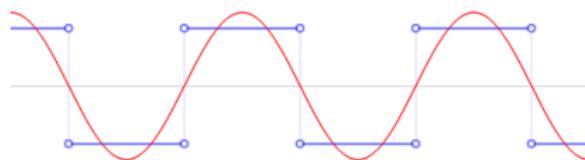


Жан Батист Жозеф Фурье  
1768–1830

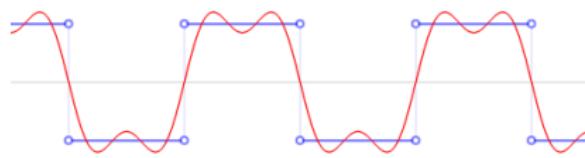
Любая функция одной переменной может быть представлена взвешенной суммой синусов и косинусов разных частот



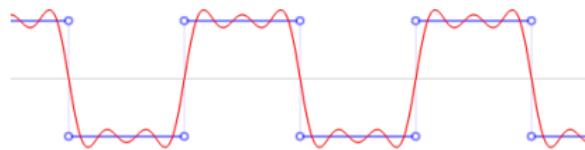
$N = 0$



$N = 1$

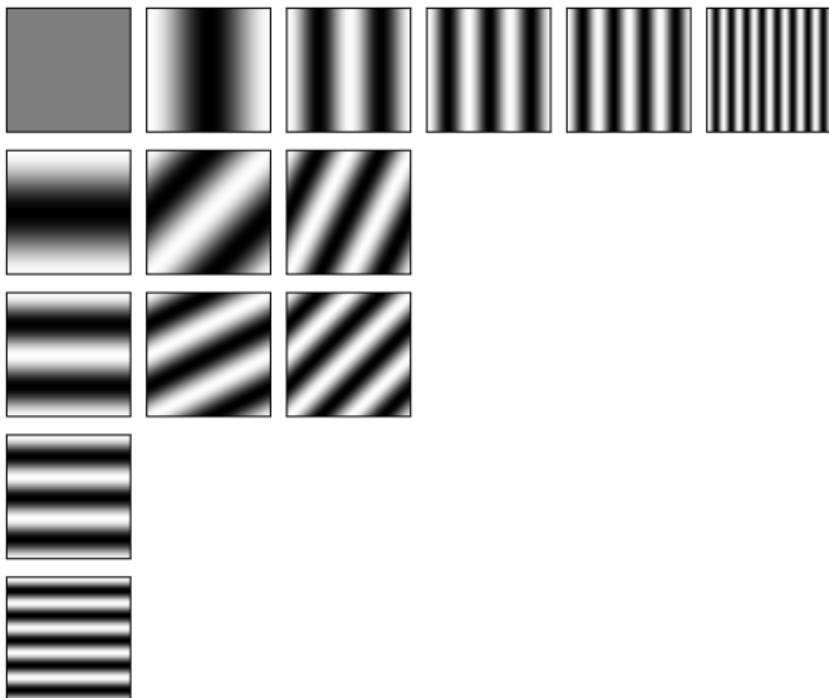


$N = 2$



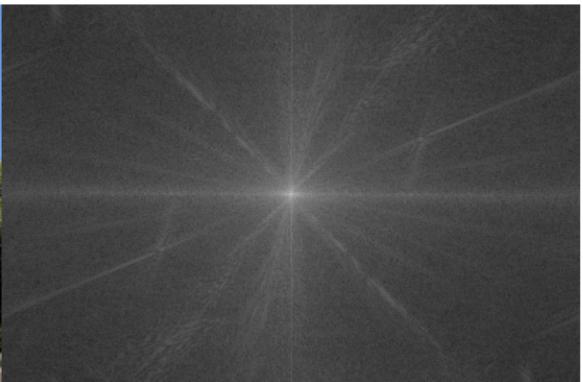
$N = 3$

# Базис Фурье для изображений





```
freq = log(1 + abs(fftshift(fft2(img))))
```



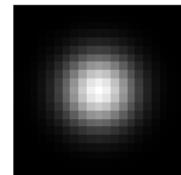
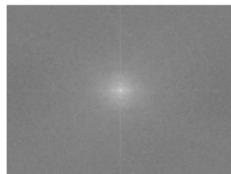
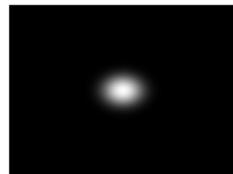
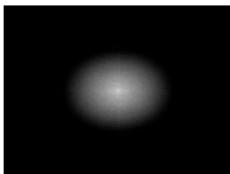
# Теорема о свертке

Преобразование Фурье свертки двух функций — это произведение преобразований Фурье этих функций

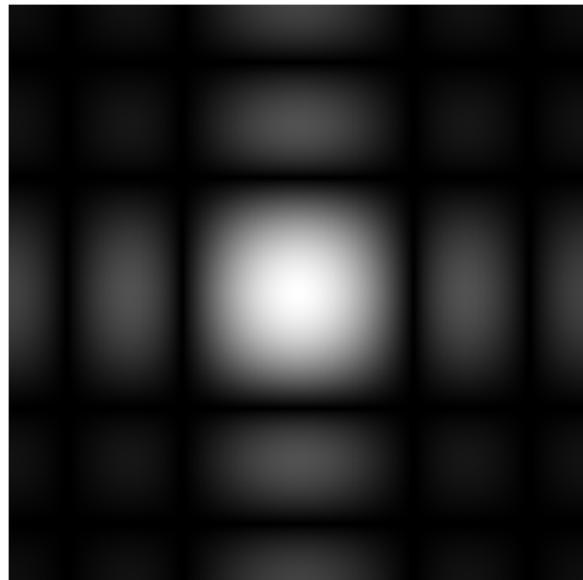
$$\mathcal{F}[g * h] = \mathcal{F}[g]\mathcal{F}[h],$$

т.е. свертка эквивалента произведению частот:

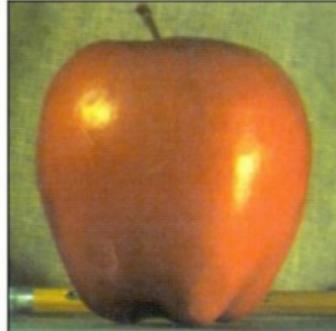
$$g * h = \mathcal{F}^{-1}[\mathcal{F}[g]\mathcal{F}[h]]$$

 $\downarrow \mathcal{F}$  $\downarrow \mathcal{F}$  $*$  $\downarrow \mathcal{F}^{-1}$ 

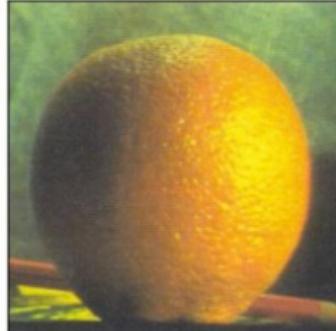
Фурье-образ box фильтра



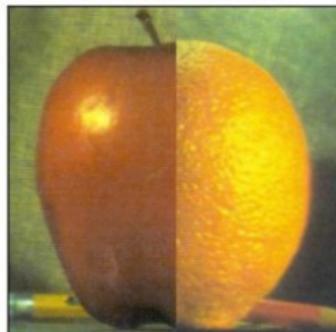
Apple



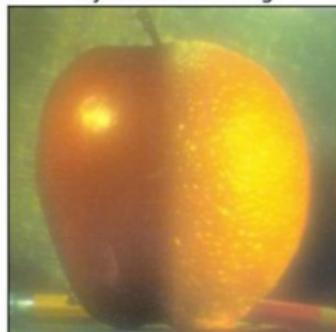
Orange



Direct Connection



Pyramid Blending



# Лапласовская пирамида

Вход:

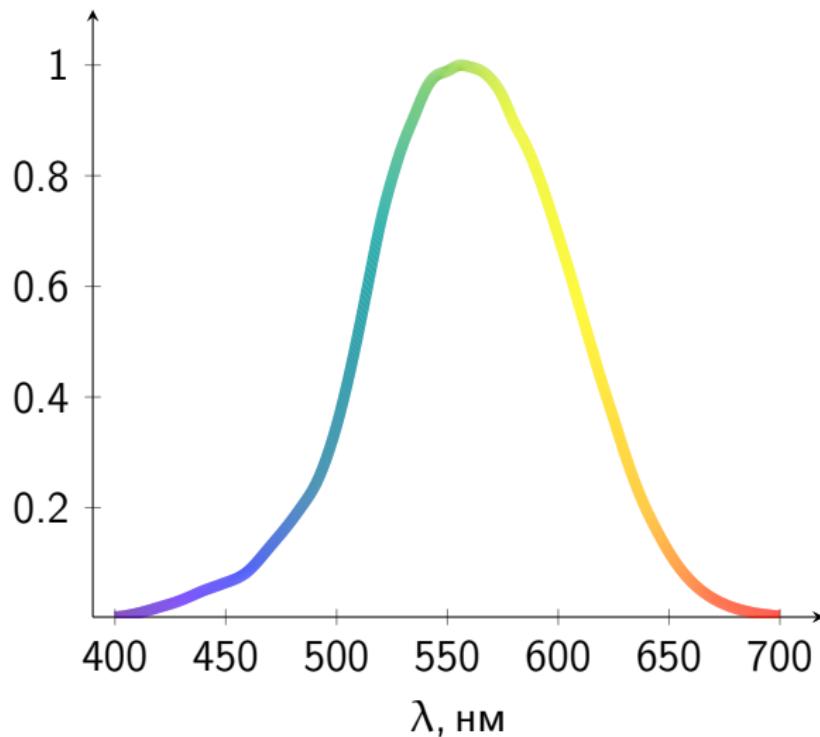
Изображения A, B, маска M

Алгоритм склейки:

1. Строим лапласовские пирамиды LA и LB из изображений A и B
2. Строим гауссовскую пирамиду GM
3. Комбинируем лапласовские пирамиды:  
$$LS = GM * LA + (1 - GM) * LB$$
4. Строим результат из пирамиды LS



яркость



$$\text{RGB} \rightarrow \text{YCbCr}$$

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$$

$$C_b = -0.169 \cdot R - 0.331 \cdot G + 0.500 \cdot B + 128$$

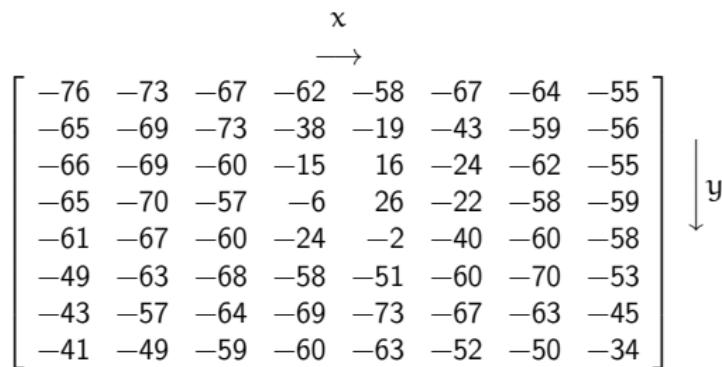
$$C_r = 0.500 \cdot R - 0.419 \cdot G - 0.081 \cdot B + 128$$



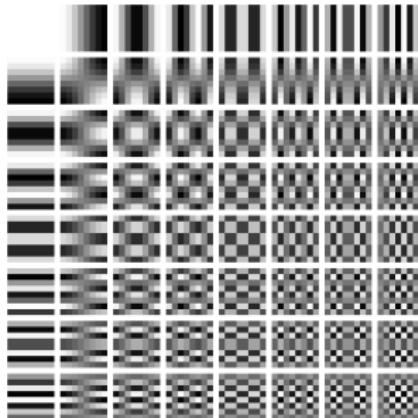
# Сдвиг на 128

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

исходное изображение



# Дискретное косинусное преобразование



$$\begin{bmatrix} & \xrightarrow{u} & \\ -415.3 & -30.1 & -61.2 & 27.2 & 56.1 & -20.1 & -2.3 & 0.4 \\ 4.4 & -21.8 & -60.7 & 10.2 & 13.1 & -7.0 & -8.5 & 4.8 \\ -46.8 & 7.3 & 77.1 & -24.5 & -28.9 & 9.9 & 5.4 & -5.6 \\ -48.5 & 12.0 & 34.1 & -14.7 & -10.2 & 6.3 & 1.8 & 1.9 \\ 12.1 & -6.5 & -13.2 & -3.9 & -1.8 & 1.7 & -2.7 & 3.1 \\ -7.7 & 2.9 & 2.3 & -5.9 & -2.3 & 0.9 & 4.3 & 1.8 \\ -1.0 & 0.1 & 0.4 & -2.4 & -0.8 & -3.0 & 4.1 & -0.6 \\ -0.1 & 0.1 & -1.0 & -4.1 & -1.1 & -0.1 & 0.5 & 1.6 \end{bmatrix} \downarrow v$$

амплитуды

# Квантование амплитуд

$$\begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}$$

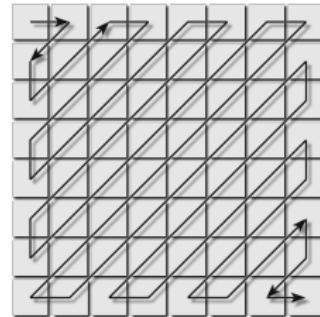
матрица квантования

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

квантованные амплитуды

# Сжатие квантованных амплитуд

$$\begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$



Кодирование серий (run-length encoding), затем  
кодирование Хаффмана

# Алгоритм JPEG

1. Конвертируем изображение в пространство YCbCr
2. Уменьшаем разрешение цветовых компонент в 2 раза
3. Делим изображение на блоки  $8 \times 8$ , вычитаем 128
4. Для каждого блока:
  - 4.1 применяем дискретное косинусное преобразование
  - 4.2 квантуем коэффициенты (амплитуды)
  - 4.3 сжимаем квантованные коэффициенты

## Резюме

Преобразование Фурье — мощный инструмент анализа изображения и поведения фильтров

С помощью Фурье-анализа можно теоретически обосновать применимость лапласовской пирамиды для фотомонтажа

Алгоритм сжатия изображений JPEG работает на основе дискретного косинусного преобразования, квантования и сжатия амплитуд