



НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Факультет компьютерных наук
Департамент программной инженерии
Курсовая работа
Кодер мультиспектральных изображения на
основе дискретно косинусного преобразования

Выполнил студент группы 204ПИ

Обыденков Д.О.

Научный руководитель:

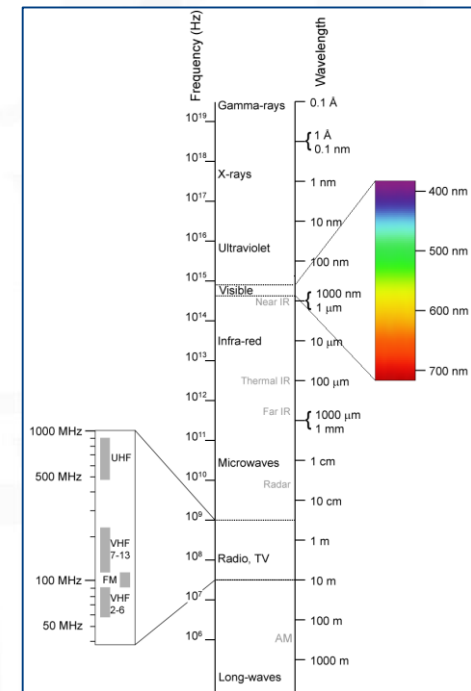
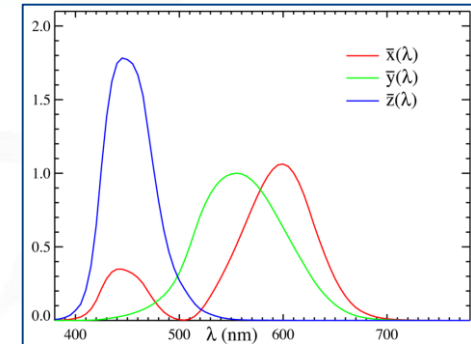
доцент департамента программной инженерии, к.т.н.

Ахметсафина Р.З.

Мультиспектральное изображение – изображение из несколько цветowych плоскостей, каждая из которых содержит информацию о некотором спектральном диапазоне длин волн (спектральном канале).

Спектральный канал – непрерывный относительно узкий участок спектрального диапазона, регистрируемый одним из датчиков и отображаемый на одну цветовую плоскость. Спектральный канал характеризуется положением центра поддиапазона и шириной поддиапазона.

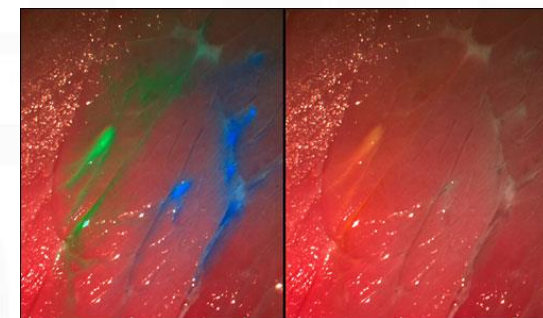
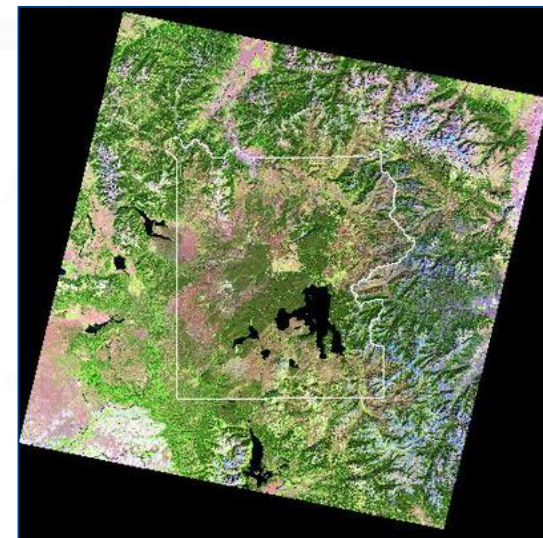
Изображения RGB – тоже являются мультиспектральными изображения с тремя частично перекрывающимися каналами. Для RGB изображений спектральными каналами являются каналы R, G, B с центрами в поддиапазонах примерно 650 нм, 540 нм, 430 нм.



Мультиспектральные системы технического зрения широкого используются в:

- дистанционном зондировании Земли для нужд
 - сельского хозяйства
 - МЧС
 - картографии
- медицине
- военном деле
- многих других областях

Основной класс решаемых задач – анализ характеристик, распознавание объектов или протяженных участков местности, поэтому к мультиспектральным камерам предъявляются достаточно высокие требования как по пространственному разрешению, так и по точности передачи спектральной информации.





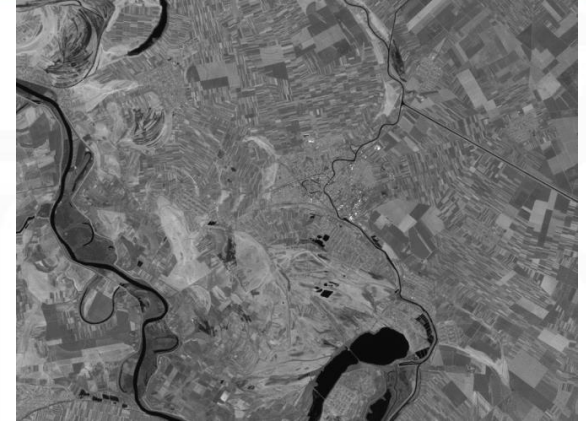
Пример мультиспектрального изображения



480 нм



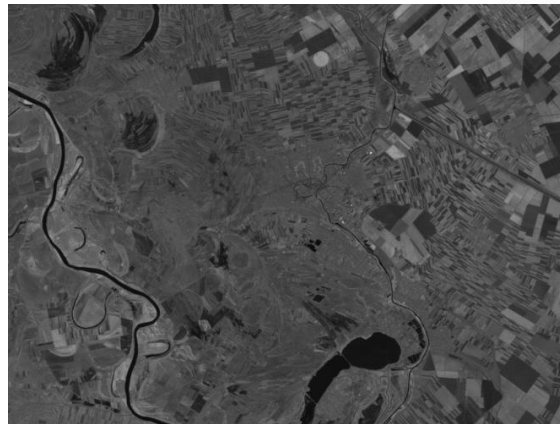
670 нм



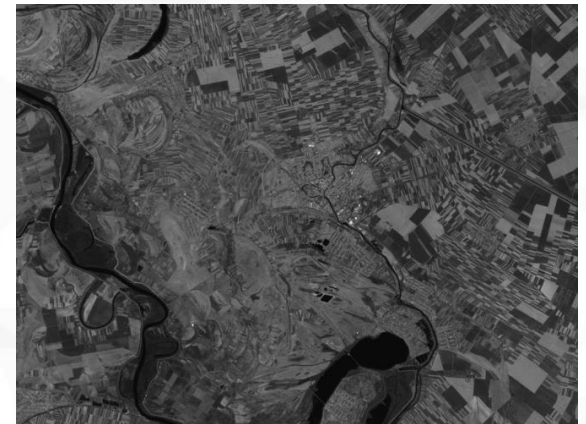
1650 нм



550 нм



820 нм



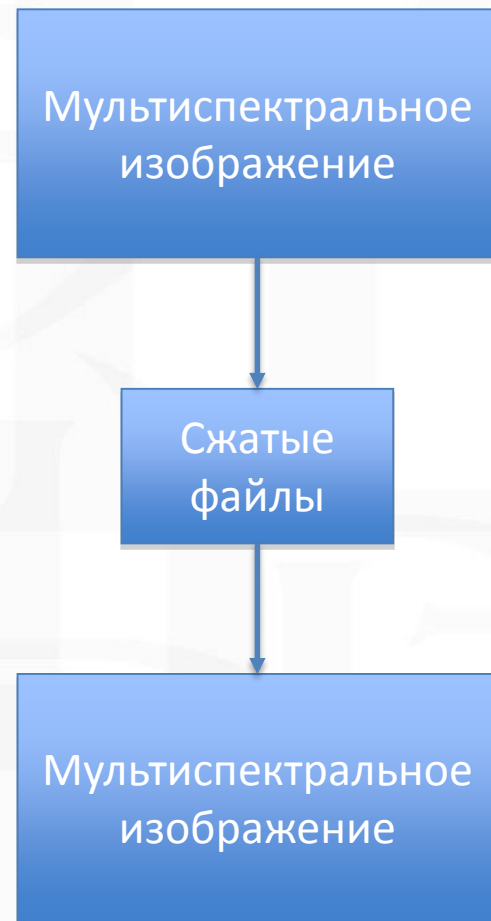
2200 нм

Мультиспектральное изображение должно быть представлено серией файлов изображений одинакового размера в формате BMP, имеющих разрешение 8 бит на пиксель. Каждый файл – спектральная плоскость, содержащая информацию об интенсивности излучения определенной длины волны в данной точке снимка.

Цель – написать программу, позволяющую:

- Сжимать мультиспектральное изображение в набор файлов меньшего размера
- Восстанавливать исходное мультиспектральное изображение

При сжатии приемлемы потери.





Задачи

- Изучить предметную область (алгоритмы сжатия изображений с потерями)
- Проанализировать методы, позволяющие выполнить сжатие изображения
- Выбрать метод сжатия изображения и разработать алгоритм для реализации
- Создать приложение, провести тестирование
- Разработать техническую документацию



Выбор метода решения задачи

Методы сжатия изображений с потерями делятся на:

- Уменьшение площади цветового пространства
- Цветовая субдискретизация
- Кодирование преобразованием
- Фрактальное кодирование

Для разработки был выбран метод, основывающийся на стандарте сжатия изображений JPEG. Он включает в себя преобразование пиксельных данных с помощью дискретно-косинусного преобразования и кодирование данных с помощью кода Хаффмана. Данный метод сочетает в себе максимальную степень сжатия с приемлемой скоростью работы.

JPEG
ISO/IEC 10918



Описание алгоритма

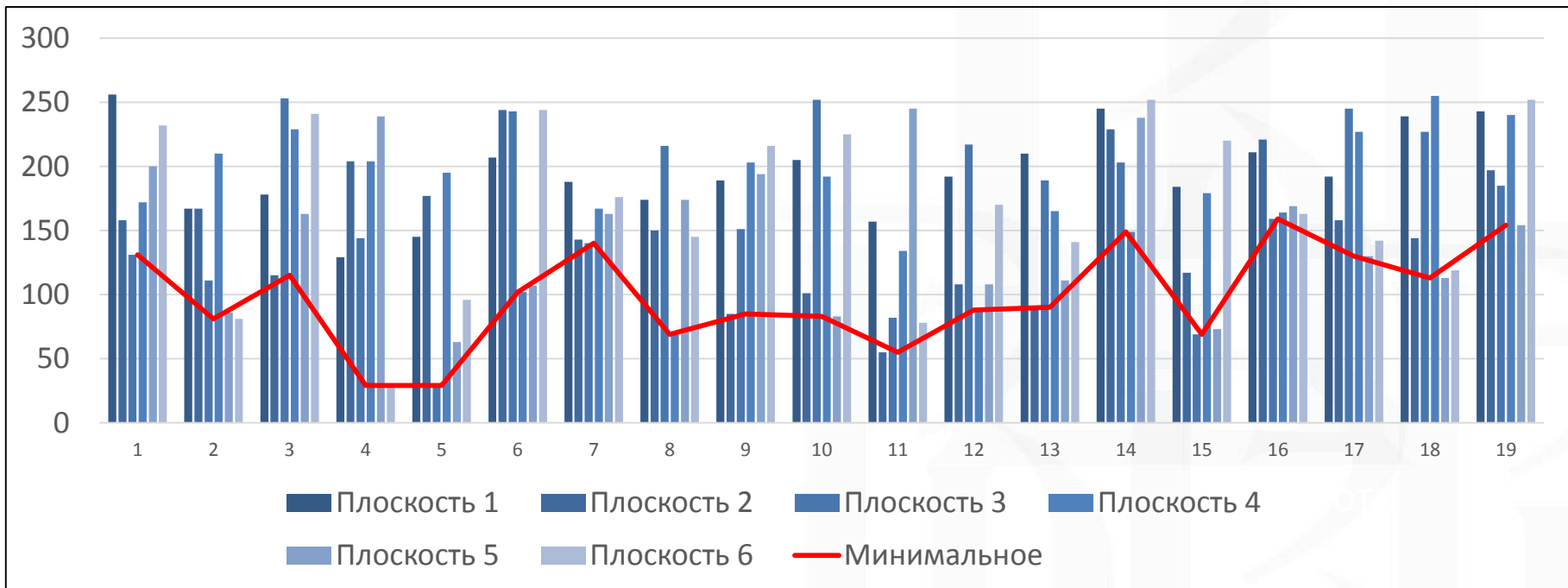
1. На вход принимается мультиспектральное изображение, состоящее из N спектральных плоскостей
2. Выделяется минимальная спектральная плоскость
3. Из каждой спектральной плоскости вычитается минимальная спектральная плоскость
4. К спектральной плоскости применяется прореживание (опционально)
5. Каждая спектральная плоскость разбивается на блоки размером 8×8
6. Каждый блок подвергается дискретно-косинусному преобразованию
7. Каждый блок квантуется в соответствии с выбранным уровнем качества;
8. Каждая спектральная плоскость кодируется кодами Хаффмана
9. На выходе получаем набор закодированных спектральных плоскостей



Минимальная спектральная плоскость

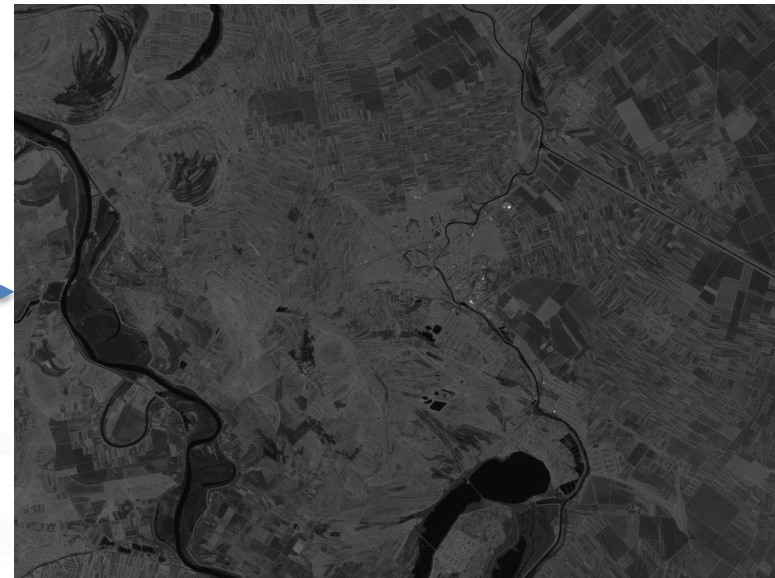
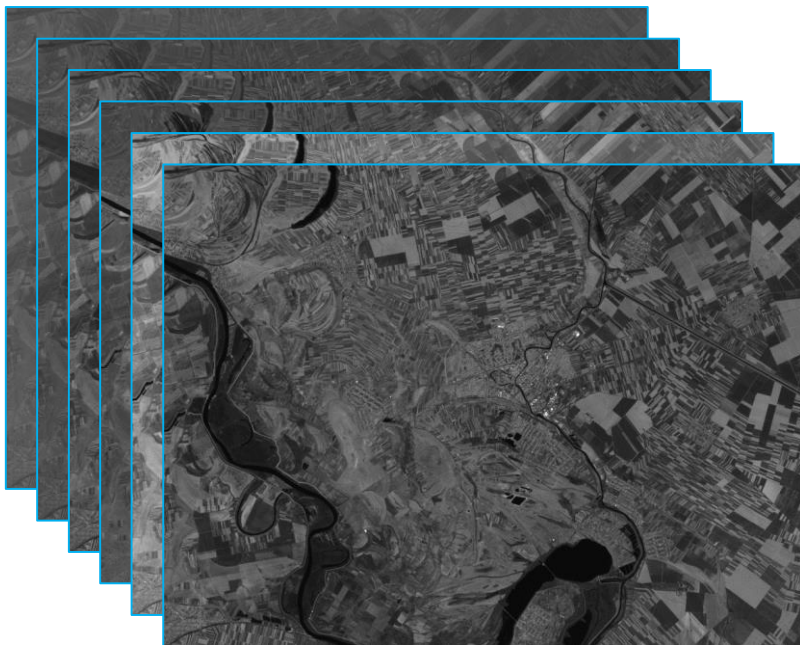
Значения минимальной спектральной плоскости формируются исходя из значений каждой спектральной плоскости

$$I_m[x, y] = \min_i(I_i[x, y])$$





Пример минимальной спектральной плоскости





Дискретно косинусное преобразование

Блоки 8x8 подвергаются обработке согласно следующей формуле:

$$DCT: \quad G_{u,v} = \frac{1}{4} C(u)C(v) \sum_{x=0}^7 \sum_{y=0}^7 g_{x,y} \cos \left[\frac{(2x+1)u\pi}{16} \right] \cos \left[\frac{(2y+1)v\pi}{16} \right]$$

$$IDCT: \quad g_{x,y} = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v) G_{u,v} \cos \left[\frac{(2x+1)u\pi}{16} \right] \cos \left[\frac{(2y+1)v\pi}{16} \right]$$

$$C(t) = \begin{cases} 1 & u = 0 \\ \sqrt{2} & u \neq 0 \end{cases}$$



Полученные после дискретного косинусного преобразования коэффициенты представляют собой вещественные числа. Для того чтобы привести к целочисленному типу проводится квантование коэффициентов. На данном этапе происходит потеря данных.

$$B_{u,v} = \text{round}\left(\frac{G_{u,v}}{Q_{u,v}}\right)$$

От подбора коэффициентов матрицы квантования зависит то, сколько данных будет потеряно в процессе сжатия. В данной программе используется три матрицы квантования:

- $Q = 10$
- $Q = 50$
- $Q = 90$

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

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

Исходный блок

$g =$

$$\begin{matrix} & & & & x & & & & \\ & & & & \rightarrow & & & & \\ \begin{bmatrix} -76 & -73 & -67 & -62 & -58 & -67 & -64 & -55 \\ -65 & -69 & -73 & -38 & -19 & -43 & -59 & -56 \\ -66 & -69 & -60 & -15 & 16 & -24 & -62 & -55 \\ -65 & -70 & -57 & -6 & 26 & -22 & -58 & -59 \\ -61 & -67 & -60 & -24 & -2 & -40 & -60 & -58 \\ -49 & -63 & -68 & -58 & -51 & -60 & -70 & -53 \\ -43 & -57 & -64 & -69 & -73 & -67 & -63 & -45 \\ -41 & -49 & -59 & -60 & -63 & -52 & -50 & -34 \end{bmatrix} & & & & & & & & y. \\ & & & & & & & & \downarrow \end{matrix}$$

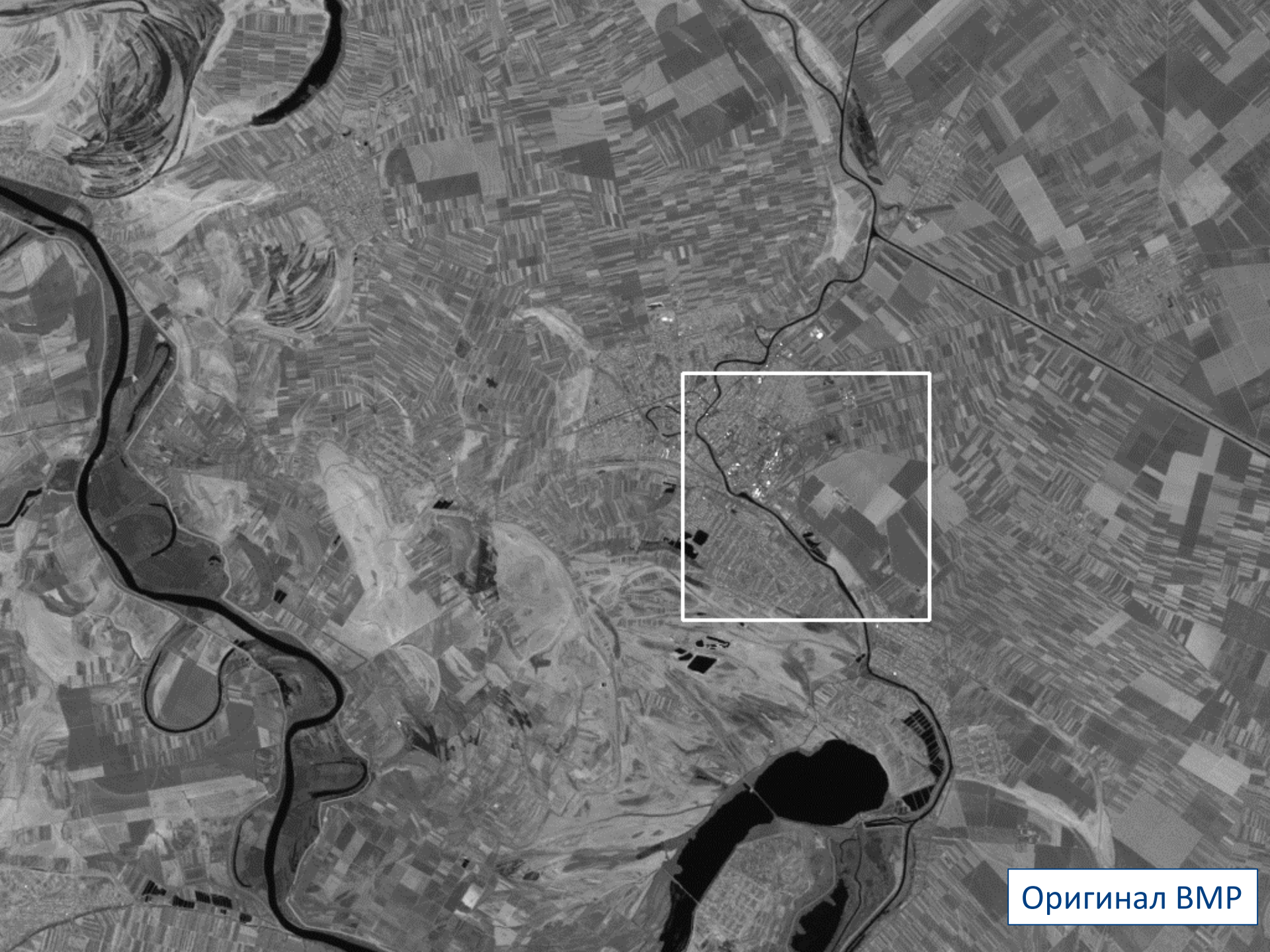
$$B = \begin{bmatrix} -26 & -3 & -6 & 2 & 2 & -1 & 0 & 0 \\ 0 & -2 & -4 & 1 & 1 & 0 & 0 & 0 \\ -3 & 1 & 5 & -1 & -1 & 0 & 0 & 0 \\ -3 & 1 & 2 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Квантованный блок

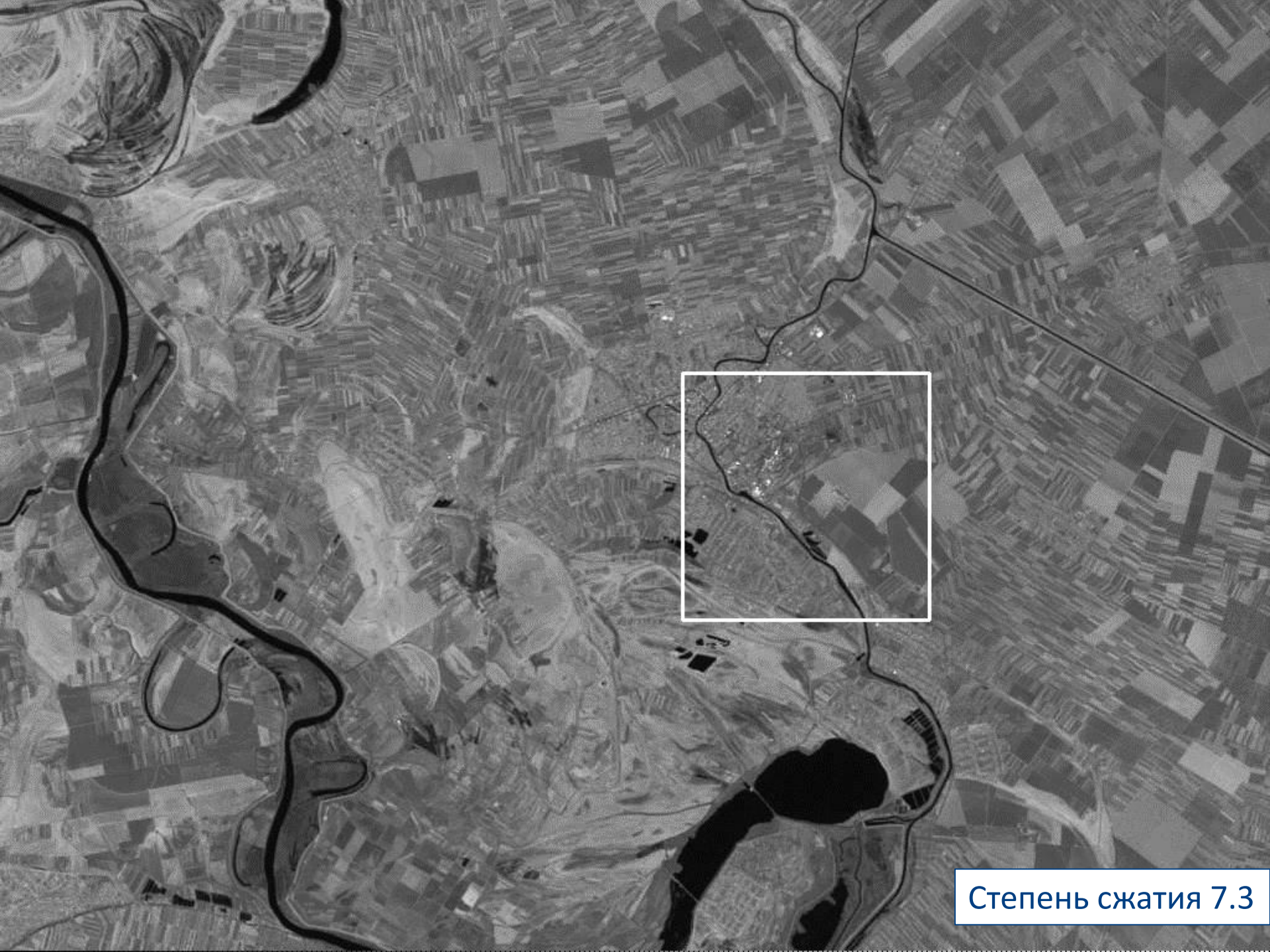
$G =$

$$\begin{matrix} & & & & u & & & & \\ & & & & \rightarrow & & & & \\ \begin{bmatrix} -415.38 & -30.19 & -61.20 & 27.24 & 56.12 & -20.10 & -2.39 & 0.46 \\ 4.47 & -21.86 & -60.76 & 10.25 & 13.15 & -7.09 & -8.54 & 4.88 \\ -46.83 & 7.37 & 77.13 & -24.56 & -28.91 & 9.93 & 5.42 & -5.65 \\ -48.53 & 12.07 & 34.10 & -14.76 & -10.24 & 6.30 & 1.83 & 1.95 \\ 12.12 & -6.55 & -13.20 & -3.95 & -1.87 & 1.75 & -2.79 & 3.14 \\ -7.73 & 2.91 & 2.38 & -5.94 & -2.38 & 0.94 & 4.30 & 1.85 \\ -1.03 & 0.18 & 0.42 & -2.42 & -0.88 & -3.02 & 4.12 & -0.66 \\ -0.17 & 0.14 & -1.07 & -4.19 & -1.17 & -0.10 & 0.50 & 1.68 \end{bmatrix} & & & & & & & & v. \\ & & & & & & & & \downarrow \end{matrix}$$

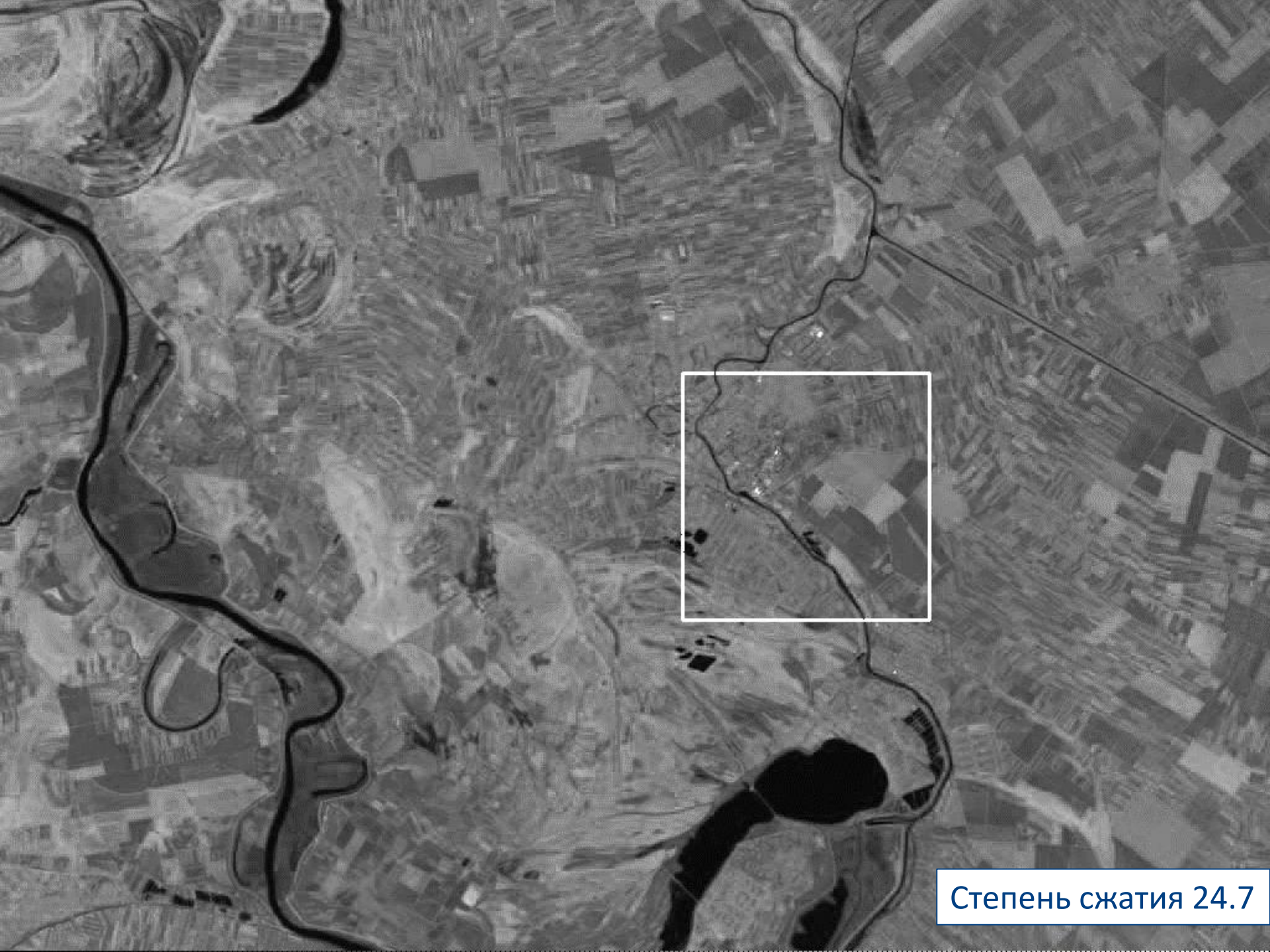
Обработанный DCT блок



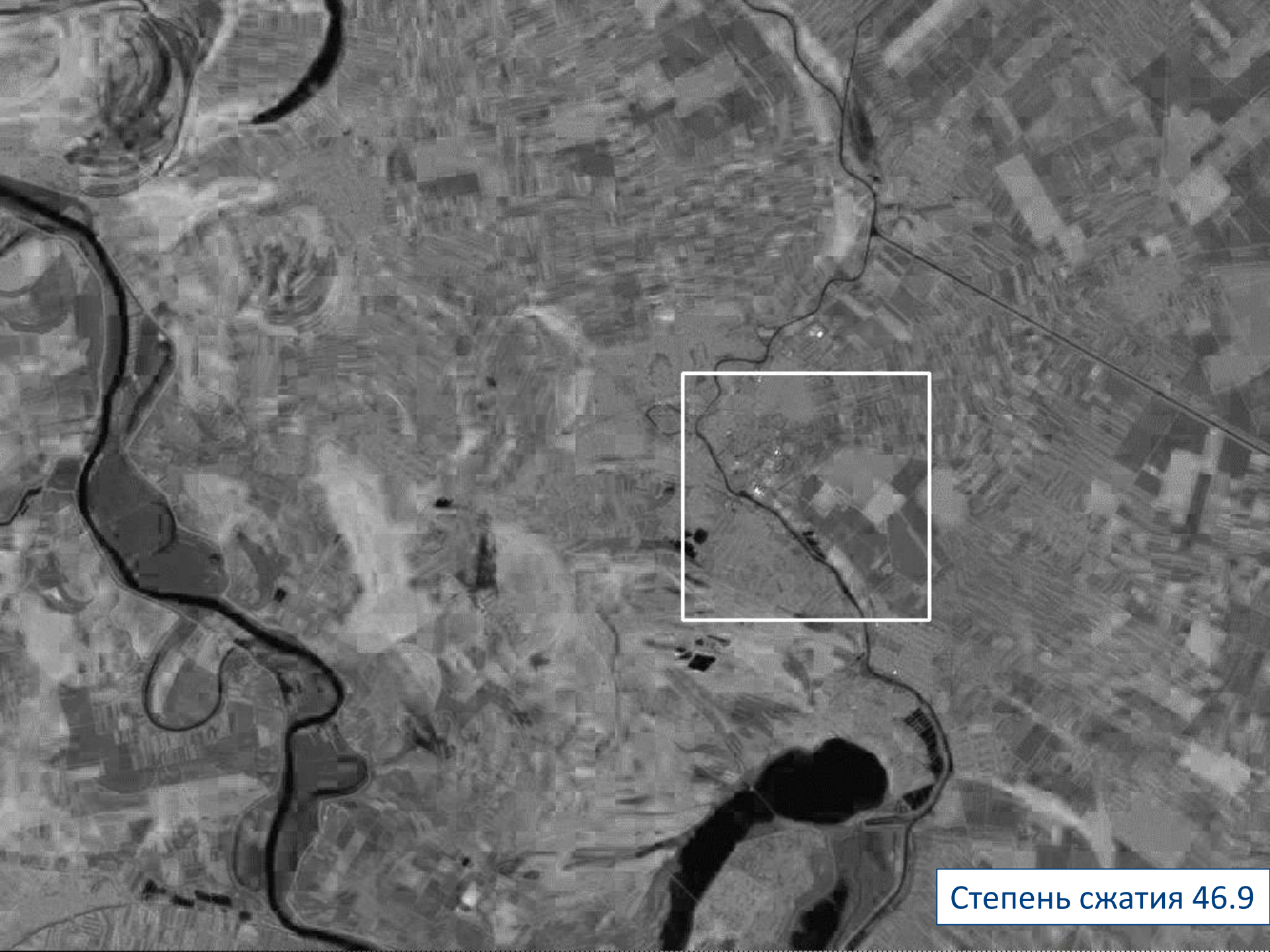
Оригинал ВМР



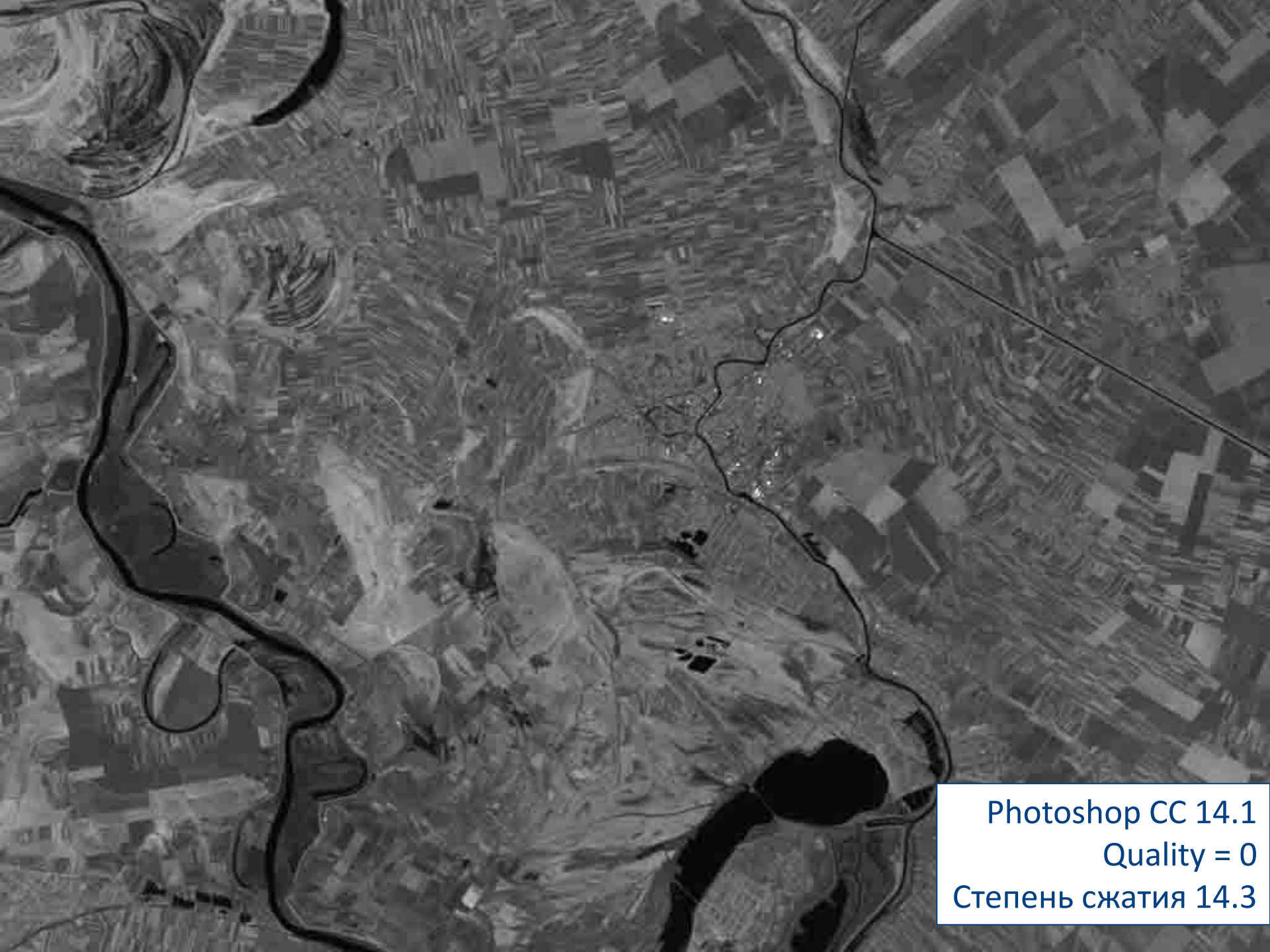
Степень сжатия 7.3



Степень сжатия 24.7



Степень сжатия 46.9



Photoshop CC 14.1
Quality = 0
Степень сжатия 14.3



Сравнение

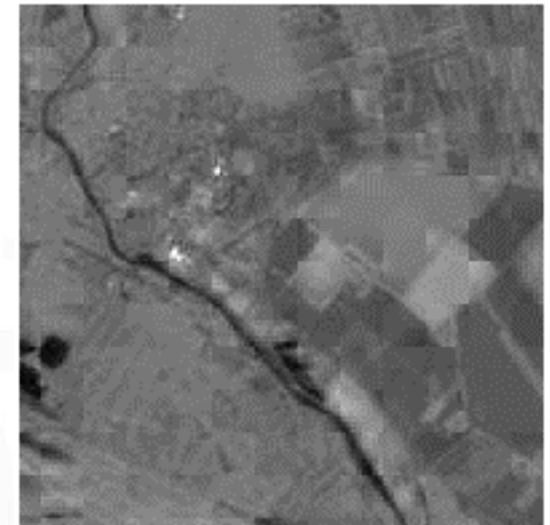


Оригинал BMP

*Степень сжатия 7.3
SNR = 39.4*



*Степень сжатия 24.7
SNR = 37.9*



*Степень сжатия 46.9
SNR = 35.0*



Критерии оценки при испытании

Были выделены следующие критерии оценки программы:

- Время сжатия мультиспектрального изображения
- Время восстановления мультиспектрального изображения
- Степень сжатия мультиспектрального изображения
- Соотношение «сигнал/шум» вычисляемое по данной формуле:

$$SNR = 20 \log_{10} \frac{P_{src}}{P_{err}}$$

$$P_{src} = \sum_{x=1}^W \sum_{y=1}^H \sum_i^N (p_{x,y,i}^{src})^2$$

$$P_{err} = \sum_{x=1}^W \sum_{y=1}^H \sum_{i=1}^N (p_{x,y,i}^{src} - p_{x,y,i}^{dst})^2$$

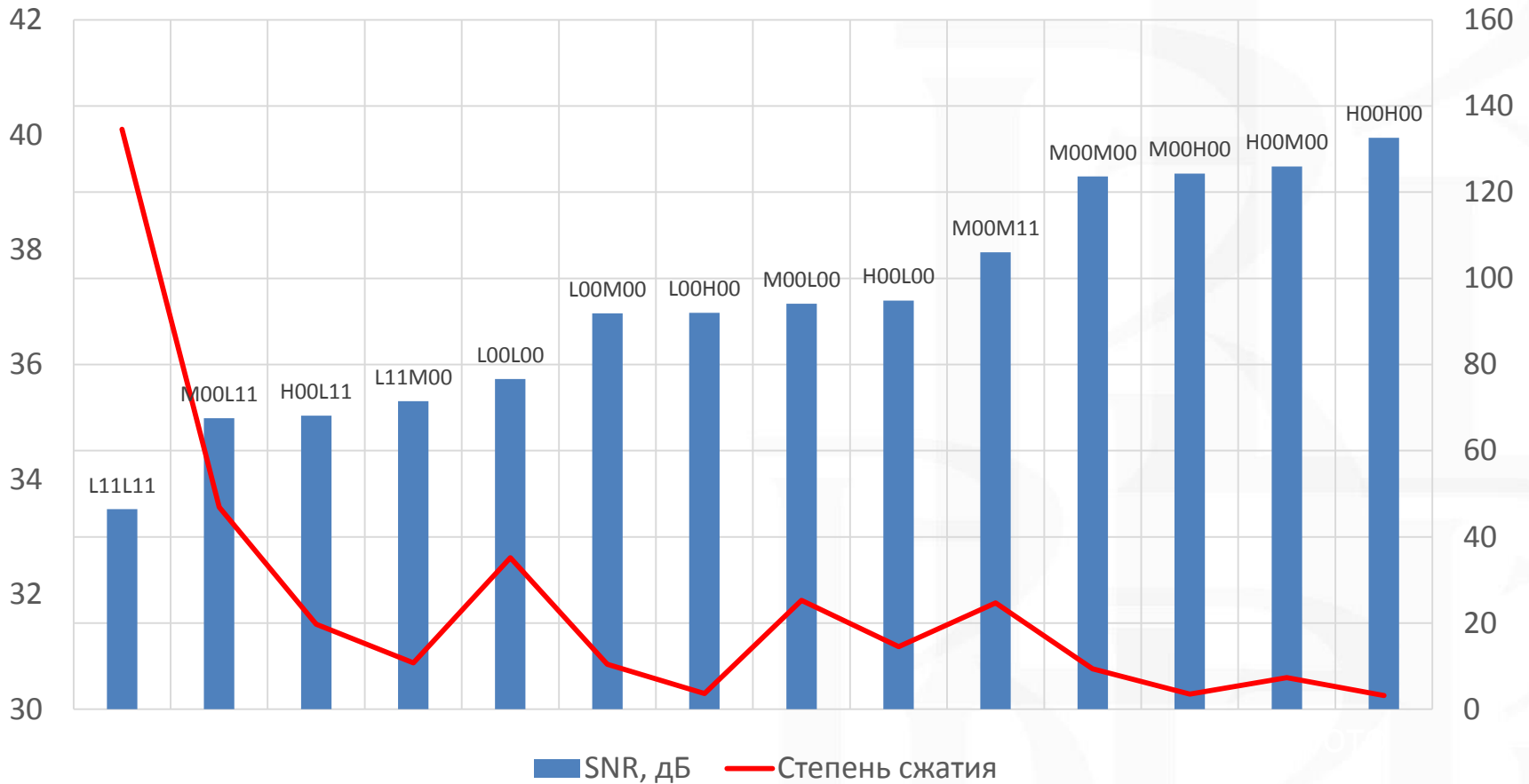


Результаты испытаний

Настройки	SNR, бД	Вес сжатого, байт	Степень сжатия	Время сжатия, мс	Время декомпрессии, мс
L11L11	33.4808	35972	134.599	6658	6898
M00L11	35.0656	103183	46.92436	9484	9873
H00L11	35.1086	245323	19.73641	9375	10010
L11M00	35.3618	450979	10.73619	22658	24580
L00L00	35.7498	137744	35.15069	25157	27313
L00M00	36.8898	464188	10.43068	25453	26838
L00H00	36.8978	1337264	3.620673	25814	27625
M00L00	37.0581	191746	25.25109	25650	27389
H00L00	37.1144	333886	14.50134	23853	29716
M00M11	37.9528	195988	24.70455	9233	10480
M00M00	39.2752	518190	9.343669	29016	28356
M00H00	39.3259	1391266	3.480137	10821	27995
H00M00	39.446	660330	7.332388	28246	30083
H00H00	39.9445	1533406	3.157543	27083	28592

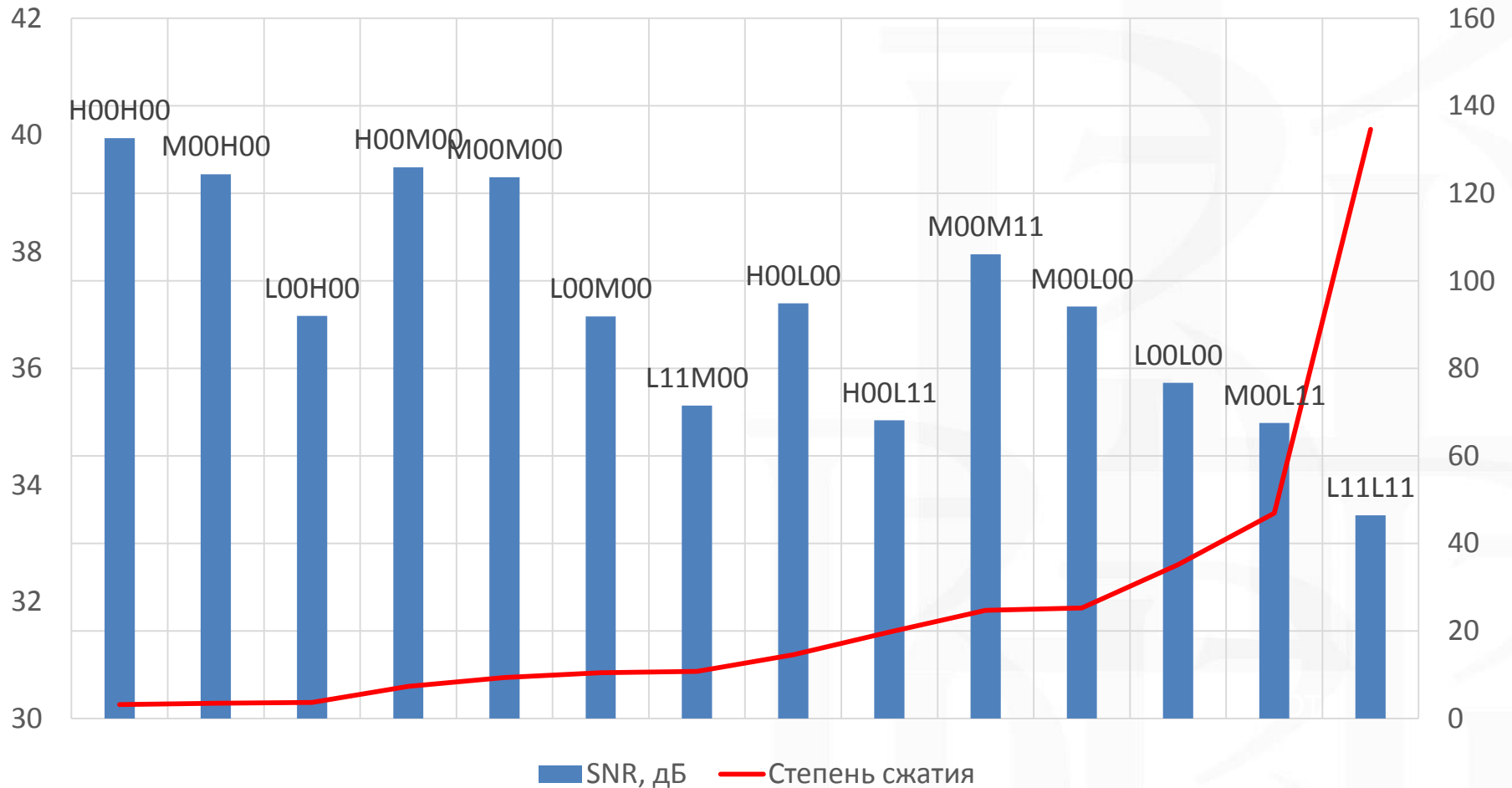


Сравнение степени сжатия с уровнем шумов



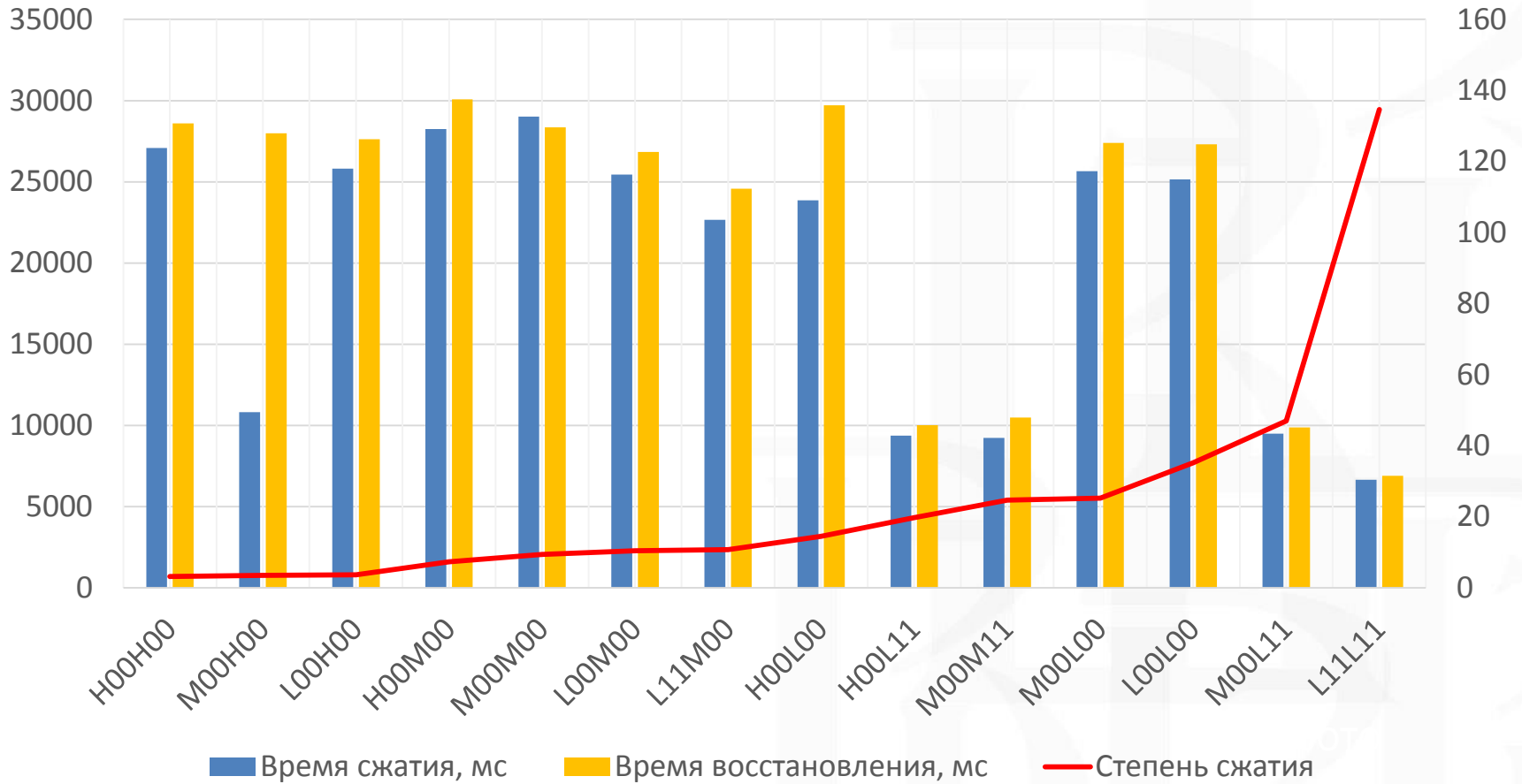


Сравнение степени сжатия с уровнем шумов





Время сжатия/восстановления



При реализации использовались следующие программы средства:

- Библиотека – C++11 STD
- GUI и работа с файловой системой – Qt 5.4





НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
УНИВЕРСИТЕТ

Спасибо за внимание!

101000, Россия, Москва, Мясницкая ул., д. 20

Тел.: (495) 621-7983, факс: (495) 628-7931

www.hse.ru