

Аппаратная реализация на базе FPGA рандомизированного метода сжатия изображений¹

О. П. Исаев

Санкт-Петербургский государственный университет

oleg.p.isaev@gmail.com

Рассмотрены традиционные подходы к сжатию изображений. Проведена оценка сложности их реализации. На основе применения парадигмы Compressive Sensing (CS) сокращения размерности массивов данных для задач сжатия изображений представлен математический аппарат получения прямых рандомизированных измерений. Выбран метод восстановления исходного изображения по измеренным данным с помощью процедуры ℓ_1 -оптимизации. Реализован аппаратный энкодер на базе CS для FPGA. Проведена сравнительная оценка занимаемой логики в реконфигурируемой логической схеме (FPGA) для стандарта JPEG и для рандомизированного метода. Показаны основные достоинства и недостатки предложенного аппаратного метода сжатия изображения.

Ключевые слова: рандомизированные измерения, ℓ_1 -оптимизация, восстановление разреженных сигналов, сжатие информации, FPGA, Compressive Sensing, DCT.

1. Введение

Теория обработки информации начала активно развиваться в начале прошлого века под растущим спросом на телемеханические системы передачи данных и управления. С появлением вычислительных машин и развитием локальных сетей, а затем и Всемирной паутины непрерывно росли новые требования к обработке текстов и аудиовизуальной информации. Так, желание максимально сократить объем передаваемых данных, привело к разработке новых алгоритмов сжатия, а в месте с ними и новых технологий, как, например, потоковая передача данных, получившая широкое распространение, особенно в мультимедийном контенте.

Увеличение вычислительной мощности позволило обрабатывать еще больший объем данных, и, как следствие, появилась возможность предоставлять пользователям более качественные услуги. В настоящее время намечается тенденция перехода от обычного телевизионного стандарта к стандарту видео высокой четкости. Вместе с тем, растущая популярность 3-D технологий требует еще большей

¹©О. П. Исаев, 2010

производительности систем и еще большей пропускной способности каналов передачи данных. Но высокая производительность требует больших энергозатрат, что неприемлемо для мобильных систем, для которых понятие “энергоэффективность” стоит не на последнем месте. Таким образом, для решения задач на современном уровне нужны комплексные подходы, включающие в себя последние технологические достижения, а также новые, более эффективные методы и алгоритмы обработки информации.

В статье рассматриваются как традиционные подходы в области сжатия изображений, так и предлагается новая система, поддерживающая метод сжатия информации на базе принципов Compressive Sensing (CS). Новый принцип работает на прямом получении сжатого сигнала из входного потока данных и реконструкции его через оптимизацию, основанную на ℓ_1 -норме. В соответствии с парадигмой CS вместо кодирования кадров по некоторому закону можно напрямую прореживать сигнал с помощью случайной матрицы A размерности $M \times N$, состоящей из равновероятно выбранных $\{-1, 0, +1\}$. Для такого кодирования достаточно провести ряд сложений и одно деление, для нормализации выходной величины. Так как размерность блоков, с которыми ведется работа, обычно кратна двум в степени n , где $n \in \mathbb{R}^n$, то деление можно заменить на операцию сдвига.

Целью представляемой в этой статье работы была проверка гипотезы о том, что при аппаратной реализации энкодера по сравнению с традиционными методами будет значительный выигрыш по занимаемому количеству вентиля в кристалле, а также по частоте его работы. Для этого необходимо получить разреженное представление тестового сигнала и алгоритм для его последующего восстановления. Кроме того, нужно решить вопрос о формировании такой матрицы измерения A , чтобы такое восстановление было возможным. Наконец, нужно подобрать оптимальный алгоритм реконструкции прореженных данных, дающий минимальную ошибку восстановления при максимальном коэффициенте сжатия, а также дать оценку объема вентиля для реализации такого рандомизированного энкодера.

2. Трансформирующее кодирование

При сжатии изображения принимают во внимание тот факт, что соседние пиксели с высокой вероятностью будут подобны друг другу либо отличаться на небольшую величину. И если к изображению применить математическое преобразование, которое уберет взаимную корреляцию между соседними пикселями, то в результате размерность массива данных сократится. И чем больше подобных элементов будет иметь массив данных, тем лучше сожмется информация.

Существует несколько подходов, которые производят декорреляцию соседних пикселей. Каждый из них имеет свои достоинства и недостатки. Рассмотрим основные методы:

- дискретно-косинусное преобразование (DCT) — частный случай дискретного преобразования Фурье, обладает хорошей способностью убирать взаимную корреляцию из массивов данных;
- преобразование Кархунена-Лозэ — наилучший метод концентрации энергии, который, однако, требует подбора и хранения коэффициентов для каждого изображения отдельно, что является трудоемким и затратным по месту, занимаемому на хранение коэффициентов, нивелируя преимущества перед DCT;
- преобразование Уолша-Адамара — уступает по эффективности сжатию с использованием DCT, однако, для реализации достаточно простых операций в виде сложений, вычитаний и, иногда, одного сдвига вправо;
- преобразование Хаара — вейвлет преобразование [1].

Наиболее широкое распространение из представленного списка получил подход на основе DCT в силу его достаточно простой реализации и высокой скорости работы. В результате этого DCT активно используется в стандартных алгоритмах сжатия аудиовизуальной информации. В частности, оно используется в таких известных стандартах, как MP3, JPEG и MPEG.

В настоящее время все большее распространение получает новый формат сжатия видео H.264. Он основан на двумерном дискретном вейвлет преобразовании, для его реализации необходимы

динамические вычисления коэффициентов “полосовых” фильтров, как в случае Кархунена-Лоэвэ, он достаточно сложен в реализации и накладывает дополнительные требования на аппаратные ресурсы. На практике реализации алгоритмов H.264 чаще всего представлены на базе того или иного специализированного сигнального процессора (DSP). Несмотря на все недостатки, этот алгоритм сжимает эффективнее, чем MPEG и считается форматом сжатия следующего поколения.

2.1. Дискретно-косинусное преобразование (DCT)

При работе с 2-D-изображениями используется двумерное дискретно-косинусное преобразование DCT-II:

$$G_{k_1, k_2} = \sum_{i=0}^{N_1-1} \sum_{j=0}^{N_2-1} c_i c_j X_{i,j} \cos \frac{(2i+1)k_1\pi}{2N_1} \cos \frac{(2j+1)k_2\pi}{2N_2}, \quad (1)$$

где $k_l = 0, \dots, N_l - 1$, $l = 1, 2$, $X_{i,j}$ — входной двумерный поток данных, коэффициенты c_i и c_j :

$$c_i = \begin{cases} \frac{1}{\sqrt{n}} & \text{при } i = 0, \\ \sqrt{\frac{2}{n}} & \text{при } i \neq 0, \end{cases} \quad c_j = \begin{cases} \frac{1}{\sqrt{n}} & \text{при } j = 0, \\ \sqrt{\frac{2}{n}} & \text{при } j \neq 0, \end{cases}$$

где n — размерность матрицы $(X_{i,j})$.

Выражение (1) является симметричным, т. е. объем данных поданных на вход, равен объему данных, принятых на выходе. Для упрощения вычислений на практике используют одномерное DCT, примененное к массиву данных, но два раза. Это вызвано тем, что DCT преобразования в математическом смысле можно интерпретировать как переход от одного базиса к другому, а в геометрическом — как вращение. А два вращения матрицы размерности $N \times N$ выполнить проще, чем одно вращение матрицы размерности $N^2 \times N^2$.

Отметим, что для обработки звука используют модифицированную одномерную версию дискретно-косинусного преобразования (MDCT) на базе DCT-IV. Отличие от DCT-II состоит в том, что на выходе порождается объем данных в два раза меньший, чем было подано на вход [2].

2.2. Сжатие изображения на основе DCT

Используя DCT-II преобразование, без дополнительных мер можно сократить объем информации за счет устранения избыточной информации (в лучшем случае до двух раз). При этом обратное преобразование IDCT-II полностью восстановит исходные данные. Такое сжатие называют сжатием без потерь [1].

Для сжатия изображения с потерями проделывают ряд дополнительных действий. Они стандартные и включают в себя следующие шаги:

- квантование;
- энтропийное сжатие;
- упаковку данных в стандартный контейнер.

Цель квантования состоит в получении наименьшего количества ненулевых компонентов в матрице преобразования при заданном качестве изображения. Для его выполнения пользуются таблицей квантования. Стандартом JPEG заданы четыре стандартные таблицы, коэффициенты которых подобраны экспериментально. Их значения можно менять по своему усмотрению. После квантования коэффициенты округляются до ближайшего целого числа. При правильно проведенной операции квантования наибольшие числа останутся в левом верхнем углу, а в правом нижнем будут преимущественно нули. Практическим результатом является то, что убирается лишняя информация, неразличимая для глаз. После проводится Zig-Zag сканирование, в результате которого матрица размером $k_1 \times k_2$ преобразовывается в вектор-строку. Причем вытягивание в строку производится так, чтобы после преобразования данные будут убывать слева направо.

Далее берутся только ненулевые компоненты и упаковываются архиватором, чаще всего используется метод Хаффмана. Более прогрессивные программы сжатия могут использовать арифметическое кодирование с несколькими проходами по дереву.

Какие таблицы использовались и какое сжатие было применено, указывается в стандартном файле-контейнере, в который упаковывается служебная информация в заданном формате и сжатое изображение. Если при сжатии использовались стандартные таблицы, то контейнер может содержать меньше информации, что позволяет дополнительно сэкономить место в файле.

Для большей оптимизации аппаратных и вычислительных ресурсов перед устранением избыточности проводят выравнивание входного потока, в процессе которого данные разбивают на блоки. Для стандарта JPEG блоки фиксированы размером 8×8 пикселей. В результате сокращается вычислительная сложность и экономится внутренняя память, используемая для хранения промежуточных результатов преобразования. Кроме того, появляются дополнительные пути и возможности для распараллеливания обработки изображений, так как все блоки могут преобразовываться независимо друг от друга.

Типовая архитектура энкодера имеет вид, представленный на рис. 1.



Рис. 1: Типовая архитектура энкодера.

2.3. Аппаратные решения для трансформирующего кодирования

Трансформирующее кодирование на базе DCT достаточно просто реализовать аппаратно. В формуле (1) можно заметить, что коэффициенты c_i и c_j , а также значения косинусов для каждого шага можно вычислить один раз и затем результаты вычислений хранить

в постоянной памяти устройства. Отсюда имеем, что для вычисления DCT нам необходимо иметь входной поток данных, каждый элемент в котором умножается на константу, соответствующую текущему шагу, и результат суммируется с полученными значениями на предыдущих шагах.

Количества элементарных операций для различных методов вычисления DCT представлены в табл. 1 [3].

Таблица 1: Ресурсы для реализации DCT.

Метод	Умножитель	Сдвиг	Сумматор
Прямое одномерное DCT	64	0	56
Chen	16	0	26
Zoeffler	11	0	29
Zoe Huang	12	5	29
AAN	13	6	29
AAN unscaled	5	0	29

Ресурсы, необходимые для реализации на FPGA JPEG энкодера для наиболее “легкого” метода AAN unscaled, представлены в табл. 2. Максимальная частота такого энкодера составляет 80 МГц.

Таблица 2: Ресурсы для реализации метода AAN unscaled.

Операция	Умножитель	BRAM	Регистр	LUTs
Выравнивание на входе	0	5	180	160
2-D-DCT	10	1	580	1000
Квантование	1	0	35	24
Кодер переменной длины	0	1	660	700
ROM с табличными данными	0	1	0	0
ИТОГО:	11	8	1455	1884

Типовая структурная схема видекамеры, с реализацией алгоритма JPEG сжатия на базе FPGA, будет иметь вид, представленный на рис. 2.

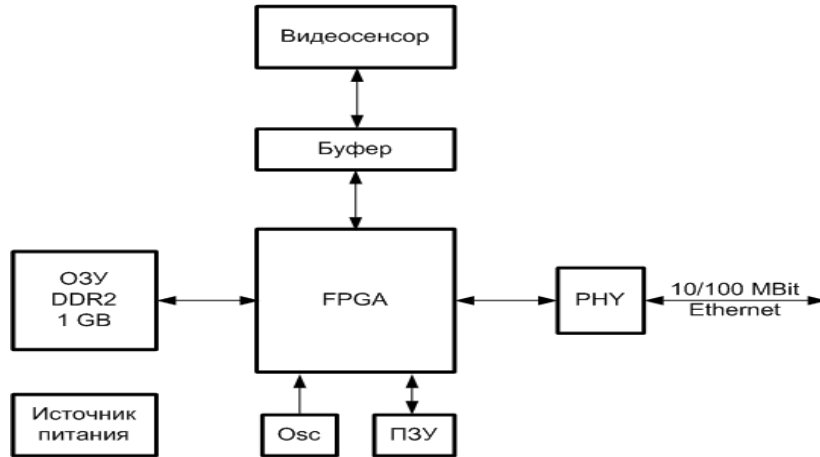


Рис. 2: Структурная схема видеокамеры.

3. Рандомизированное сжатие изображения

В качестве основы рассмотрим описанную в [6] однопиксельную камеру. Она реализована на базе микроконтроллерной системы управления микрозеркалами, линзы, светодиода и аналогово-цифрового преобразователя (АЦП).

В качестве аппаратной реализации матрицы измерений используется цифровой микрозеркальный модуль (DMD) с разрешением 1024×768 от “Texas Instruments”, предназначенный для видеопроекторов. Каждое микрозеркало имеет свою SRAM-память и может позиционироваться независимо от других микрозеркал в два положения $+12$ или -12 градусов по горизонтали. Через объектив изображение фокусируется на зеркальной матрице и отражается либо в сторону двояковыпуклой линзы, либо отклоняется от нее. В свою очередь, линза фокусирует прореженный пучок света на фотодиод и через 12-ти битный АЦП суммарная информация передается в цифровой канал связи [7].

Общая структурная схема устройства на базе однопиксельной камеры имеет вид представленный на рис. 3.

Для того, чтобы восстановление изображения было возможным, необходимо провести $t \ll N \cdot M$ измерений, при этом происходит позиционирование зеркал для каждого шага измерения по псевдо-

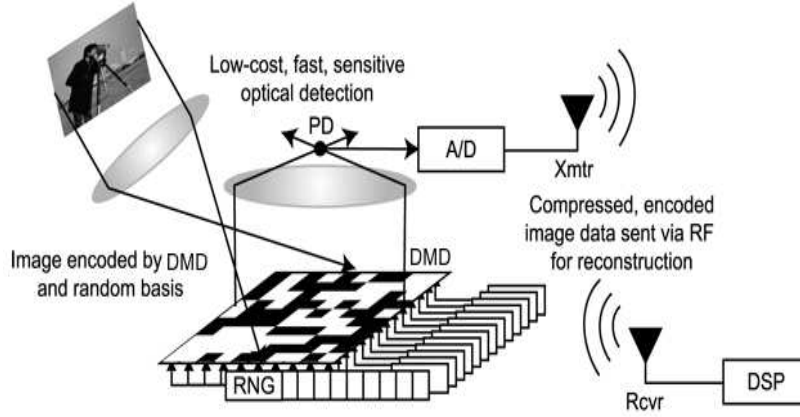


Рис. 3: Структурная схема однопиксельной камеры в общем виде.

случайному закону. В частности, авторы статьи [7] использовали распределение Бернулли для генерации матрицы измерения.

3.1. Прореживание сигнала

Опишем систему, в которой проводится m измерений y при воздействии m случайных матриц ϕ на изображение X . Одно наблюдение можно записать как:

$$y_i = \frac{1}{k} \phi_i^T x, \quad (2)$$

где $i \in [1, m]$, размерность вектора ϕ равна $K = N \cdot M$. Значение K определяется размером изображения X , где N и M — размерность строки и столбца данного изображения соответственно. Вектор x получен из X путем преобразования матрицы размерности $N \times M$ в вектор размерности K . Из (2) видно, что в результате i -го измерения получаем просто число y_i .

Запишем полный вектор измерений в следующем виде:

$$Y = (y_1, y_2, y_3, \dots, y_m)^T, \quad (3)$$

В формуле (2) ϕ_i^T — строки общей матрицы измерений Φ , каждый элемент которой принимает значения '0' или '1'. Правила вы-

бора стабильной матрицы измерений рассмотрены в статье [8, 9]. В частности, в [9] описана возможность формирования матрицы Φ некоторым случайным или псевдослучайным методом с некоторой фиксированной вероятностью распределения P_a , которая, впрочем, может изменяться по внешнему требованию.

Запишем формулу получения вектора измерений в общем виде:

$$Y = \Phi x, \quad (4)$$

где матрица Φ имеет размерность $m \times N$ и равна: $\Phi = (\phi_1, \phi_2, \dots, \phi_m)^T$

Выражение (4) описывает трансформацию входного вектора x в Y , или другими словами, производит переход из пространства \mathbb{R}^K в пространство \mathbb{R}^m , где $m < K$. Таким образом, можно утверждать, что Y является разреженным (sparse) представлением исходного изображения X .

3.2. Восстановление изображения

Для того чтобы восстановить изображение из массива, представленного в разреженном пространстве, необходимо перейти из одного базиса в другой. Для методов на основе DCT достаточно решить обратную задачу трансформирующего кодирования (IDCT).

В случае системы уравнений (4) задача не может быть решена классическими методами из-за несовпадения соответствующих размерностей матриц Φ , x и Y .

Для того, чтобы восстановить сигнал $x \in \mathbb{R}^K$ из разреженного $Y = \Phi x \in \mathbb{R}^m$, где $m \ll K$, необходимо решить задачу выпуклой оптимизации:

$$(P1) \quad \min \|x\|_{\ell_1} \quad \text{при условии} \quad Y = \Phi x. \quad (5)$$

В статье [10] рассмотрены семь проблем, решение которых сводится к решению подобного рода задач. Они делятся на два класса. Первый класс использует в своей основе решение задачи линейного программирования (LP), второй решает задачу выпуклой оптимизации второго порядка (SOCPs). Для решения задач LP используется прямо-двойственный алгоритм следования вдоль пути, а для задач нелинейного программирования используются барьерные логарифмические штрафные функции. Алгоритмы, предназначенные для решения всех семи проблем, представлены в виде

библиотеки для среды MATLAB и ее можно скачать по адресу <http://www.acm.caltech.edu/11magic>.

Для решения задачи P1 известен метод Basis pursuit (преследование базиса), который заключается в поиске вектора x_0 , дающего минимум ℓ_1 -нормы. При этом решение x_0 будет представлять декодированное изображение.

Если входной сигнал представляет собой 2-D изображение, то лучший результат восстановления будет давать альтернативная модель, использующая один из SOCPs алгоритмов.

Пусть $X_{i,j}$ представляет собой один пиксел изображения X размерности $N \times N$, где i и j номера строки и столбца соответственно, и определен операторами:

$$D_{h;ij}X = \begin{cases} X_{i+1,j} - X_{ij} & i < n, \\ 0, & i = n, \end{cases} \quad D_{v;ij}X = \begin{cases} X_{i,j+1} - X_{ij}, & i < n, \\ 0, & i = n, \end{cases}$$

$$D_{ij}X = \begin{pmatrix} D_{h;ij}X \\ D_{v;ij}X \end{pmatrix}. \quad (6)$$

D_{ij} в формуле (6) можно интерпретировать как класс дискретных градиентов для цифрового изображения X . Полная вариация (total variation) X — это просто сумма значений дискретного градиента в каждой точке, и ее можно записать так:

$$TV(x) = \sum_{ij} \sqrt{(D_{h;ij}X)^2 + (D_{v;ij}X)^2} = \|D_{ij}X\|_2$$

Для того, чтобы восстановить изображение, необходимо решить задачу минимизации полной вариации x :

$$(TV_2) \quad \min TV(x) \quad \text{при условии} \quad \|Y - \Phi x\| \leq \epsilon. \quad (7)$$

В отличие от (5) в формулу (7) введено дополнительное ограничение на поиск решения, заданное квадратичной погрешностью.

3.3. Реализация модели рандомизированного сжатия изображения

Принимающая сторона должна знать, какие именно матрицы были использованы в процессе измерения. Это требование накладывает существенные ограничения на реальную систему. Задача

легко решается при использовании псевдослучайного генератора, который работает по одинаковому закону и одинаковых начальных условиях на обеих сторонах. Очевидно, что реализация адаптивного генератора, который на каждом шаге динамически будет подбирать закон измерения под конкретное изображение, является не оптимальной задачей. Это вытекает в силу того, что совместно с измерениями необходимо будет передавать и начальные условия, для синхронизации работы псевдослучайного генератора. На практике эти начальные условия будут иметь размерность, превышающую размерность самих данных, что приведет к дополнительной нагрузке на канал передачи данных и к неоптимальному его использованию.

Для простоты моделирования примем, что матрица Φ порождена некоторым случайным законом, значения ее строк для каждого шага измерения будем хранить в памяти вычислительной машины и по мере надобности передавать их значения декодеру для корректного восстановления изображения.

В качестве примера рассмотрим черно-белые картинки, с градацией серого равной восьми битам. Это связано с тем, что, во-первых, глаз человека более чувствителен к яркости изображения, чем к ее цветовой составляющей, а во-вторых, с черно-белыми картинками проще работать.

Для проведения измерений необходимо иметь следующие компоненты:

- псевдослучайный генератор, с изменяемым законом распределения плотности вероятности следования '1';
- кэш-память для хранения измеряемой матрицы x размерности $N \times M$;
- кэш-память для хранения результата измерений Y ;
- сумматор.

Апробация алгоритмов сжатия и восстановления была проведена в среде MATLAB. Разреженное представление входного сигнала реализовано в виде отдельной функции `CS_Encoder`. Для декодирования информации реализован восстанавливающий изображение декодер `CS_decoder`, основанный на решении проблемы TV минимизации с ограничением на квадратичную ошибку, реализация которого взята из библиотеки " ℓ_1 -magic", упомянутой выше.

3.4. Экспериментальные данные по CS сжатию

На первом этапе экспериментов была поставлена цель проверить работоспособность алгоритмов сжатия и восстановления на практике, а также подтвердить верность предположения о способе выбора матрицы измерения Φ . Работа проводилась с числами с плавающей запятой.

На рис. 4 представлен результат работы алгоритма сжатия. Восстановление проводилось при пороге $\varepsilon = 0.01$. При его уменьшении до 10^{-3} шум PSR уменьшался на $10dB$. А если порог сократить еще на порядок до значения $\varepsilon = 10^{-4}$, то значение PSR сокращался еще на $10dB$. По представленным результатам можно сделать вывод, что зависимость ошибки восстановления от порога точности восстановления ε — логарифмическая.



Рис. 4: Оригинал и три восстановленных изображения 100% , 40% и 20%, при $P_a = 50\%$.

Алгоритм на основе Compressive Sensing является алгоритмом сжатия с потерями и при сжатии и восстановлении одного и того же изображения всегда будет присутствовать нерегулярная ошибка. То есть снижение ошибки при уменьшение порога носит в общем-то условный характер. В качестве подтверждения этого факта проведем серию из трех экспериментов по сжатию и восстановлению изображения, по 100 раз в каждой серии. При этом серии отличаются только количеством измерений m , а плотность вероятности P_a следования единиц в матрице измерения и точность восстановления — фиксированы. Результаты эксперимента представлены на рис. 5–7.

Из графиков видно, что чем больше число измерений, тем выше вероятность получить картинку лучшего качества в следствие меньшей флуктуации ошибки относительно некоторого среднего значения. В табл. 3 представлены численные результаты для одного случайно выбранного сжатого изображения из каждой серии эксперимента.

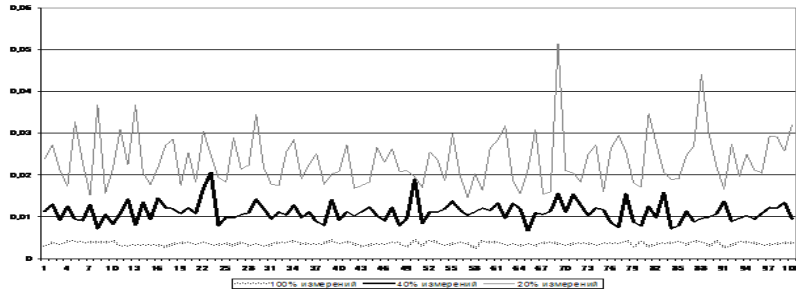


Рис. 5: Среднеквадратичная ошибка при $P_a = 10\%$.

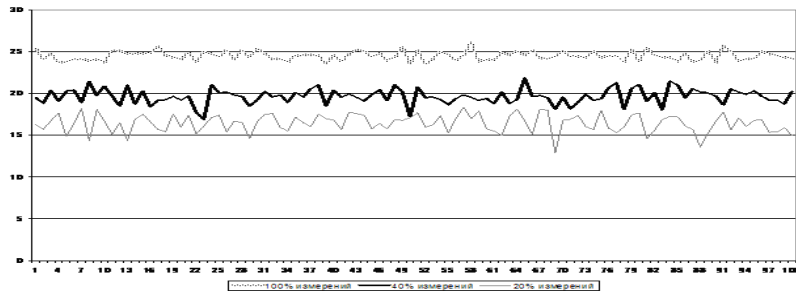


Рис. 6: Пиковое значение Сигнал/Шум при $P_a = 10\%$.

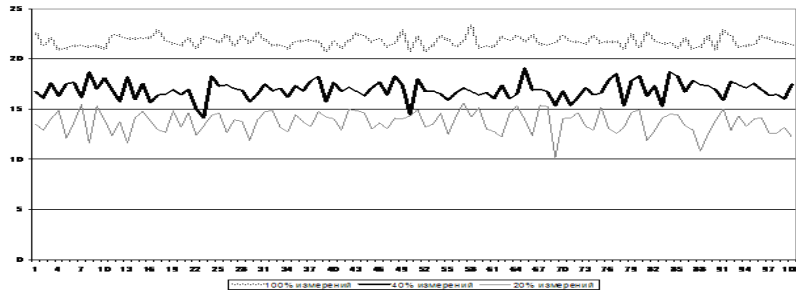


Рис. 7: Отношение Сигнал/Шум при $P_a = 10\%$.

Таблица 3: Значение основных параметров при $P_a = 50\%$.

Размер изображения, <i>pix</i>	32×32	32×32	32×32
Погрешность восстановления, ε	0.01	0.01	0.01
Среднеквадратичная ошибка	0.0014	0.0067	0.0112
Пиковое отношение сигнал/шум, <i>dB</i>	28.5235	21.7098	19.5096
Отношение сигнал/шум, <i>dB</i>	25.7745	18.9609	16.7607
Количество измерений, <i>m</i>	1024	409	204
Вероятность распределения, P_a	50%	50%	50%

Далее отметим, что изменение значения P_a в большей степени влияет на размерность измерений, чем на точность. Тем не менее в минимальных и максимальных значениях P_a точность восстановления заметно снижается. И третий, очевидный параметр, влияющий на точность восстановления, — это число измерений: чем их больше, тем она выше.

На втором этапе экспериментов решалась задача перехода от арифметики с плавающей точкой к фиксированной, так как при использовании чисел с двойной точностью сжатия изображения не происходит. Наоборот наблюдается увеличение объема информации и чем больше значение P_a , тем больше места она занимает. При упаковке выходных данных обычным архиватором он дает небольшой коэффициент сжатия порядка 15-20%. При сокращении числа наблюдений до 50%, получается размер файла соизмеримый с входным. При сжатии архиватором удавалось его уменьшить на те же 15-20%. Информация восстанавливалась с большей точностью при понижении порога ε , также было определено, что уменьшение P_a сокращает размерность данных измерения и до какого-то предела практически не влияет на качество восстановления.

В результате округления резко увеличилась ошибка восстановления. Уменьшение ε менее, чем 10^{-3} перестало влиять на точность восстановления. В общем случае был сделан эмпирический вывод, что декодеру не важно, какой размерности будет число, главное чтобы пропорция между измерениями сохранялась наиболее точно. В результате значимым фактором оказался максимальный и минимальный порог белого. При максимальном пороге измерение не должно выходить за разрядную сетку, а при минимальном пороге его значение не должно быть слишком маленьким, иначе те-

ряется информация о градациях серого. При работе с картинкой с широким динамическим диапазоном, часть информации необратимо теряется и необходимы специальные меры для его расширения после реконструкции сигнала. В этом случае, вероятность распределения единиц в матрице измерения P_a стала иметь зависимость от размера блоков, на которые было разбито изображение.

Для моделирования был реализован переход от чисел с плавающей запятой к целочисленной арифметике и проведен ряд экспериментов. Результат для 40% измерений при различных разбиениях на блоки представлен в табл. 4.

Таблица 4: Значение основных параметров при делении изображения на блоки различной величины перед сжатием.

Размер изображения, pix	256×256	256×256	256×256
Размер блоков, pix	8×8	32×32	64×64
Погрешность восстановления, ϵ	0.01	0.01	0.01
Среднеквадратичная ошибка	0.0473	0.0344	0.0171
Пиковое отношение сигнал/шум, dB	13.2497	14.6304	17.6626
Отношение сигнал/шум, dB	9.5324	10.9131	13.9454
Количество измерений, m	40%	40%	40%
Вероятность распределения, P_a	20%	10%	3%
Коэф. сжатия совместно с архиватором, раз	2.67	2.72	2.65

Как видно из табл. 4 наилучший результат по отношению Сигнал/Шум, дает метод, использующий разбиение на блоки по 64×64 пикселя.

На рис. 8 слева представлено оригинальное тестовое изображение, с которым велась работа, и его гистограмма справа. Полный размер изображения составляет 256×256 пикселя. Как видно из рис. 9 по сравнению с оригиналом происходит сужение динамического диапазона, что и показывает гистограмма данного восстановленного изображения. После ручной обработки и расширения динамического диапазона субъективное качество картинки улучшается рис. 10.

3.5. Аппаратная реализация CS энкодера

Для того, чтобы реализовать аппаратный энкодер, необходимо иметь модули, выполняющие следующие операции:

- выравнивание входного потока;
- проведение измерений;
- энтропийное сжатие;
- упаковка данных в контейнер.

Выравнивание входного потока подразумевает первичную подготовку данных для сжатия изображения, т. е. получение изображения с сенсора, размещение его в оперативной памяти и поставка упаковщику информации с разбиением его на блоки из диапазона 8×8 , 16×16 , 32×32 или 64×64 пикселей.

После этого проводится прореживание сигнала путем его измерения. Причем, для решения этой задачи необходим только сумматор и некоторое количество операций сдвига, количество которых зависит от размера блоков, на которые разбито изображение. Количество тактов измерения также зависит от размера этих блоков. Чем они меньше, тем выше должна быть вероятность P_a , и чем они

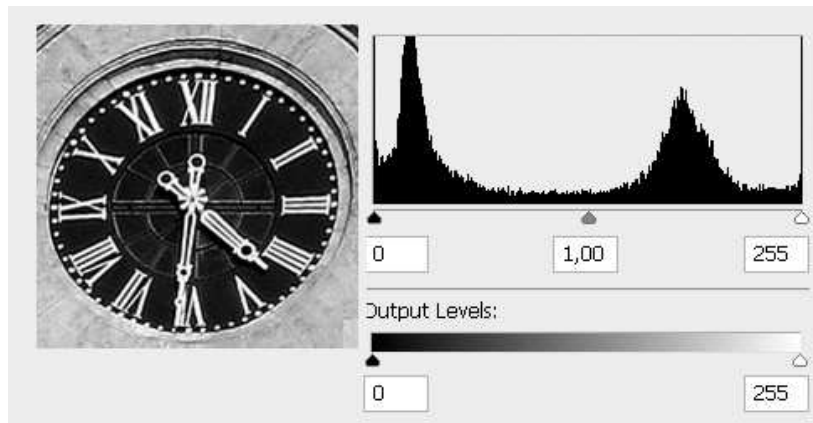


Рис. 8: Оригинальное изображение размером 256×256 пиксела и его гистограмма.

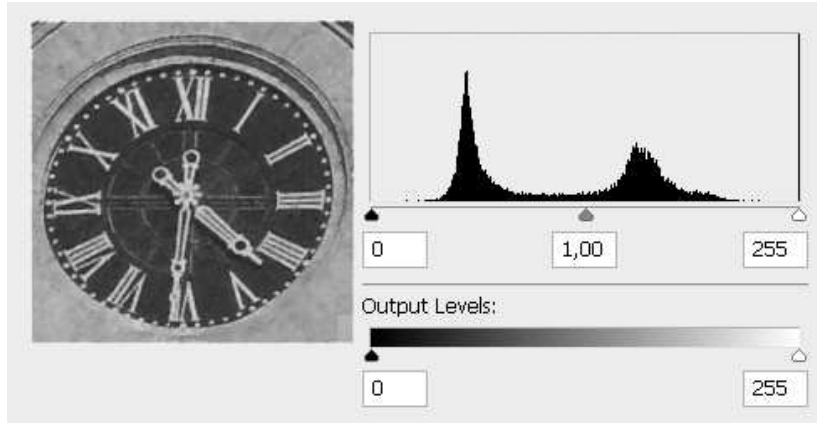


Рис. 9: Восстановленное изображения и его гистограмма при 40% измерениях и разбиении его на блоки по 64×64 пиксела.

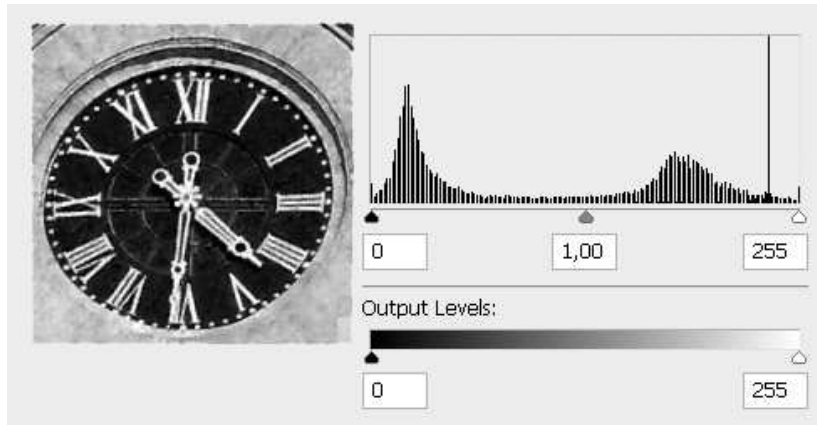


Рис. 10: Восстановленное изображение после ручной установки уровней, полученное после 40% измерений и разбиении его на блоки по 64×64 пиксела.

больше, тем меньшее значение будет у P_a . Так, если блок 8×8 , то для успешного восстановления картинке с минимальной погрешностью, вероятность P_a необходимо выбирать в районе 50%, а для блока размерностью 64×64 достаточно провести 3-5% измерений.

Несложно подсчитать, что для максимального размера блока одно измерение потребует 122 такта, а для минимального размера блока при таком же объеме информации потребуется уже 2048 тактов. При восстановлении изображения задача может иметь сложность порядка $O(K^3)$ [8], поэтому большие блоки будут восстанавливаться значительно дольше, чем малые. Однако, чем больше блок, тем выше качество его восстановления.

Для дополнительного увеличения степени сжатия уже разреженной информации можно применить архивацию без потери информации. Затем обработанный массив необходимо упаковать в контейнер со служебной информацией, которая может включать в себя такие параметры, как размер блока, количество измерений, начальные значения для псевдослучайного генератора и т. п. Отметим, что сжатая информация чувствительна к потерям. Повреждение хотя бы одного измерения, приведет к невозможности правильного восстановления соответствующего блока информации. Поэтому при передаче изображений необходимы дополнительные меры защиты данных от ошибок.

Для реализации аппаратного энкодера использовался язык VHDL. Оценка ресурсов, требуемых для реализации в FPGA для кристалла Cyclon II, при работе с одним блоком размером 64×64 пикселя:

- количество логических элементов — 165;
- количество регистров — 95;
- объем памяти — 64 кБит.

Если сравнивать с реализацией 2-D-DCT (см. табл. 2), то видно, что объем требуемой логики отличается на порядок. Предварительная оценка задержки распространения сигнала внутри кристалла по самому длинному пути, данное синтезатором, прогнозирует максимальную частоту работы ядра порядка 165 МГц.

4. Заключение

Описанная выше однопиксельная камера является, по сути, аналоговым устройством. Следовательно, для реализации подобных ей действий необходимо применять ряд мер, которые должны минимизировать вносимую погрешность измерения и позволить более

точно восстановить изображение на стороне приемника. В основном такие меры заключаются в борьбе с наводками шумов по питанию и сигнальным цепям, что достигается введением специальных цепей термомпенсации и устранением нелинейной характеристики фото диода. Также необходимо учитывать вносимые абберации в световой поток от DMD и погрешности квантования в процессе аналогово-цифрового преобразования. Если использовать вместо микрзеркал реконфигурируемую логическую матрицу, а вместо одного пикселя CMOS или CCD видеосенсор, то такая система будет лишена многих из этих недостатков. Однако, применение видеоматрицы несколько меняет идеологию, используемую в однопиксельной камере. Но к ней можно вернуться и расширить возможности с использованием стандартных аппаратных решений. Тогда в рассматриваемом контексте дрожание рук при фотографирование, возможно, может оказаться достаточно полезным свойством динамической системы.

В процессе работы, как и ожидалось, недостатком предлагаемой новой схемы является сложная процедура декодирования, которая занимает существенно большее время для работы алгоритма по сравнению с классическими. При первичном использовании получалась малая степень сжатия, дальше удалось ее повысить, но потеряв в качестве восстановленного изображения, что отразилось на значениях Сигнал/Шум и погрешности восстановления. Степень сжатия удалось получить в пределах 2–2.7 раз при более сильном прореживании, после реконструкции изображения получается низкое значение Сигнал/Шум и низкая субъективная оценка его восприятия.

В дальнейших работах планируется провести более глубокий анализ алгоритмов восстановления с целью поиска путей улучшения качества реконструкции и для их ускорения. Также поставлена задача совместного использования трансформирующего кодирования и кодирования на базе CS для сжатия видео. Есть гипотеза, что в такой конфигурации можно получить гораздо большее сжатие, чем дают современные алгоритмы.

На данный момент есть востребованная на практике задача, в которой целесообразно использовать описанный в статье подход. Необходимо разработать распределенную систему видеонаблюдения, в которой IP видеокamеры через конвертор подключены к низкоскоростной резервированной проводной сети передачи дан-

ных стандарта Е1. На систему наложены жесткие ограничения на скорость видеопотока, которая не должна превышать значения 2 МБит/с. Задача может быть решена с использованием стандарта сжатия видео на базе H.264 кодека. Но любая коммерческая реализация этого стандарта подпадает под лицензионные соглашения, что не всегда приемлемо для заказчика.

Список литературы

- [1] *Сэлмон Д.* Сжатие данных, изображений и звука. — М: Изд-во Техносфера. 2004. 368 с.
- [2] *Joebert J* Audio Compression Using Modified Discrete Cosine Transform: the mp3 Coding Standard. An undergraduate research paper submitted to the Department of Mathematics College of Science The University of the Philippines Diliman. Quezon City. Oct. 2001.
- [3] *van Dyck W., Smodic R., Hufnagl H., Berndorfer T.* High-speed JPEG Coder Implementation for a Smart Camera. — Springer-Verlag. 2006.
- [4] *Donoho D.* Compressed sensing // IEEE Trans. Inform. Theory. 2006. Vol. **52**. No. 4. Apr. P. 1289–1306.
- [5] *Santosa F., Symes W.* Linear inversion of band-limited reflection seismograms // SIAM J. Sci. Statist. Comput. 1986. Vol. **7**. No. 4. P. 1307–1330.
- [6] *Takhar D., Laska N., Michael W. and Eds.* A new compressive imaging camera architecture using optical-domain compression // Proc. of Computational Imaging IV at SPIE Electronic Imaging. San Jose. CA. Jan. 2006. [Online]. Available: <http://dsp.rice.edu/files/cs/cscam-SPIEJan06.pdf>
- [7] *Wakin B., Jason L., Marco D. and Eds* An architecture for compressive imaging // Proc. International Conference on Image Processing – ICIP 2006. Atlanta. GA. Oct. 2006. [Online]. Available: <http://dsp.rice.edu/sites/dsp.rice.edu/files/cs/CSCam-ICIP06.pdf>

- [8] *Граничин О.Н., Павленко Д.В.* Рандомизация данных и ℓ_1 -оптимизация // Компьютерные инструменты в образовании. 2010. №1. С. 5–13.
- [9] *Граничин О.Н., Павленко Д.В.* Рандомизация получения данных и ℓ_1 -оптимизация (опознание со сжатием) (Обзор) // Автоматика и телемеханика. 2010. № 11.
- [10] *Candès E., Romberg J.* ℓ_1 -magic : Recovery of Sparse Signals via Convex Programming. Caltech. Oct. 2005. [Online]. Available: <http://www.acm.caltech.edu/l1magic/downloads/l1magic.pdf>