

Московский авиационный институт  
(государственный ТЕХнический университет)

Факультет прикладной математики

Кафедра вычислительной математики и программирования

Курсовая работа по «Методам и средствам Мультимедиа»

«Методы объективной оценки потерь качества при  
компрессии цифрового видео»



Преподаватель: А. В. Крапивенко  
Студент: И. К. Никитин

Москва, 2010

# Содержание

<b>Введение</b>	<b>3</b>
<b>1 Теоретическая часть</b>	<b>5</b>
1.1 Сжатие	5
1.2 Кодеки	5
1.2.1 Cinepak	5
Векторная квантизация	6
Временное квантование	7
1.2.2 Indeo 5	8
Пространственное сжатие	9
Временное сжатие	10
Компенсация движения	10
1.2.3 MPEG-4 DivX	11
Пространственное сжатие	12
Временное сжатие	13
Двухпроходность	13
1.3 Объективные оценки	14
1.3.1 Метрика $\sup(x-y)$	14
1.3.2 Математическое ожидание разности	15
1.3.3 PSNR	16
1.3.4 Контекстные методы	18
1.4 Субъективные методы оценки качества	19
1.4.1 Шкала деградации с двумя стимулами	19
1.4.2 Непрерывная оценка качества с одним стимулом	20
1.4.3 Попарное сравнение	20
<b>2 Вычислительная часть</b>	<b>21</b>
2.1 Гистограммы яркостей	22
2.1.1 Кадр 0055	22
2.1.2 Кадр 0746	23
2.1.3 Кадр 1316	24
2.2 Субъективные оценки	25
2.2.1 Низкая степень сжатия	25
2.2.2 Высокая степень сжатия	25
2.3 Объективные оценки	26
2.3.1 Cinepak	26

2.3.2	Indeo5 . . . . .	29
2.3.3	DivX . . . . .	32
2.4	Сопоставление . . . . .	35
2.4.1	Сопоставление оценок . . . . .	35
2.4.2	Сопоставление кодеков . . . . .	38
	$\sup (x-y)$ . . . . .	38
	Разность оценок матожидания . . . . .	39
	PSNR . . . . .	40
<b>3</b>	<b>Аналитическая часть</b>	<b>41</b>
3.1	Cinepak . . . . .	41
3.1.1	$\sup$ . . . . .	41
3.1.2	Разность оценок матожидания . . . . .	41
3.1.3	PSNR . . . . .	42
3.2	Indeo5 . . . . .	42
3.2.1	$\sup$ . . . . .	42
3.2.2	Разность оценок матожидания . . . . .	43
3.2.3	PSNR . . . . .	43
3.3	DivX . . . . .	43
3.3.1	$\sup$ . . . . .	44
3.3.2	Разность оценок матожидания . . . . .	44
3.3.3	PSNR . . . . .	44
3.4	Сопоставление кодеков друг с другом . . . . .	45
3.4.1	$\sup$ . . . . .	45
3.4.2	Разность оценок матожидания . . . . .	45
3.4.3	PSNR . . . . .	46
	<b>Заключение</b>	<b>47</b>

# Введение

## Проблема

Данная курсовая работа посвящена методам объективной оценки потерь качества при компрессии цифрового видео. Цифровое видео крайне ресурсоемко. Это не сложно понять если попытаться рассчитать сколько по объему (в гигабайтах) занимает один час телевизионного вещания.

Размер кадра для телевизионного вещания составляет  $720 \times 576$ . Так же учтем, что картинка имеет три составляющих (об этом будет рассказано ниже, сразу оговоримся, что эти составляющие, не RGB, столь привычные современному человеку). Вещание ведется с частой 25 кадров в секунду.

Перемножая все эти параметры друг на друга и на время, можно получить что один час телевизионного вещания составляет 104 Гб.

$$\begin{aligned} 720 \times 576 \cdot 3 \cdot 25 \cdot 3600 = \\ = 111974400000 \text{ б} \approx 104 \text{ Гб} \end{aligned}$$

Такая цифра даже по сегодняшним меркам является недопустимой для хранения видео-данных. Отсюда возникает необходимость сжатия.

Первыми для сжатия изображений стали применяться привычные алгоритмы, которые использовались и используются в системах резервного копирования, при создании дистрибутивов и т. п. Эти алгоритмы архивировали информацию без изменений. Однако основной тенденцией в последнее время стало использование новых классов изображений. Старые алгоритмы перестали удовлетворять требованиям, предъявляемым к сжатию. Многие изображения практически не сжимались, хотя «на взгляд» обладали явной избыточностью. Это привело к созданию нового типа алгоритмов сжимающих с потерей информации. Как правило, степень сжатия и, следовательно, степень потерь качества в них можно задавать. При этом достигается компромисс между размером и качеством изображений [10]. В работе рассмотрено три различных кодека Cinepak, Indeo5, DivX.

## Цель

Целью исследования является выявление особенностей потерь при сжатии цифрового видео различными кодеками на основе методов объективной оценки.

# Задачи

Цель данной работы обусловила следующие задачи:

- 1) Раскрыть и описать теоретические принципы сжатия цифрового видео на примерах конкретных кодеков.
- 2) Провести критический обзор исследуемых кодеков (Cinерак, Indeo5 и DivX).
- 3) Описать особенности методик оценки потерь качества.
- 4) Отобрать кадры для исследования кодеков.
- 5) Привести оценки метрик для отобранных кадров и их линейную разность.
- 6) Построить график «PSNR – субъективное качество».

Объектом исследования в данной работе являются видеопоследовательности сжатые указанными кодеками. Предметом данного исследования являются методы сжатия видео.

Достижение цели исследования и решение поставленных задач обуславливает необходимость использования комплекса общенаучных теоретических (теоретический анализ, конкретизация, моделирование) и эмпирических (изучение специальной литературы, инструкций, применение численных экспериментов) методов исследования.

Характер исследования, детальность рассмотрения проблем, теоретическая значимость полученных результатов позволяет сделать вывод о том, что данное исследование может быть примером использования объективных оценок качества для анализа методов сжатия.

# 1. Теоретическая часть

## 1.1. Сжатие

Сжатие видео основано на двух важных принципах.

- пространственная избыточность;
- временная избыточность.

Различные методы сжатия по сути отличаются только способами компенсации пространственной избыточности — так называемым внутрикадровым кодированием. Чаще всего происходит преобразование (свойственно не для всех кодеков с потерей качества). После возможных преобразований происходит отбрасывание, усреднение или округление. Здесь и происходит потеря качества.

Метод сжатия видео начинается с кодирования первого кадра с помощью некоторого алгоритма компрессии изображения. Затем следует кодировать каждый последующий кадр, находя расхождение или разность между этим кадром и его предшественником и кодируя эту разность — **компенсация временной избыточности**. Если новый кадр сильно отличается от предыдущего (это происходит, например, с первым кадром последовательности), то его можно кодировать независимым образом. Далее рассматриваются различные методы сжатия видео. Способами компенсации временной избыточности они коренным образом не отличаются.

## 1.2. Кодеки

Кодек (**coder-decoder**) — это способ и реализация шифрования для записи видеоданных в файл, например в AVI, и расшифровки информации, соответственно, для воспроизведения уже записанных AVI файлов.

### 1.2.1. Cinepak

Достоинства:

- не требователен к ресурсам;
- распространен на большинства платформ.

Недостатки:

- большое число блочных артефактов для относительно не высокой степени сжатия;
- значительное время кодирования;
- устарел;

Кодек Cinepak был первоначально разработан для воспроизведения фильмов малого размера (с разрешением 320x240 со скоростью 1x (150 кбайт / с) ) на персональных компьютерах, с одной скоростью чтения компакт-дисков.

Cinepak был разработан Питером Барретом в компании SuperMac Technologies и его релиз состоялся в 1991 в составе Video Spigot, а также в 1992 в составе видео пакета QuickTime для компьютеров компании Apple. В 1993 данный кодек был включен в состав платформы Microsoft Windows. Стоит отметить, что данный кодек также использовался в игровых приставках, например: Atari Jaguar CD, Sega CD, Sega Saturn и 3DO.

Основным ее преимуществом является его чрезвычайно низкие требования к производительности компьютера. В связи с бурным развитием компьютерной техники, кодек был несколько модифицирован, в сравнении с его первоначальной реализацией. Последнее время он используется не только для фильмов с малым разрешением и продолжительностью, но при большом числе более развитых форматов встретить Cinepak на практике не просто.

Но при всем обилие современных кодеков, достаточно трудно предугадать, какие из них будут установлены к моменту просмотра на целевом компьютере. И в данном случае, проблему решает как раз Cinepak. Не трудно убедиться, что он есть на достаточно широком спектре машин

Мы думаем, что тому причина скорее политическая, нежели технологическая. Не нужно забывать, что Cinepak был одним из самых первых кодеков, и после его появления все производители платформ массово включили его в свои дистрибутивы. Альтернатив на тот момент (1991 год) было не много, и для «войны форматов» оснований не было. А по прошествии лет, кодек не стали убирать из-за его широкого распространения (ну и для обратной совместимости, конечно).

Не очень давно появилась мобильная версия Cinepak Mobile. Связано это с тем, что мобильные телефоны имеют малую производительности по сравнению с компьютерами. Безусловно — хорошее применение почти забытых технологий в новом контексте. В кодеке используются алгоритмы векторной квантизации и временного квантирования.

## Векторная квантизация

При использовании векторной квантизации изображение делится на блоки примерно равного размера. Блоки представляют собой фиксированное число векторов. Такие вектора называют *кодowymi словами*. Вводится некоторый критерий качества  $Q$ . Для блоков  $i$  и  $j$  запишем.

$$d(i, j) < Q.$$

Если дистанция между  $i$  и  $j$  меньше, чем  $Q$  в выбранной метрике, то один из этих блоков объявляется универсальным. Например, для блоков  $i, j, l, x$  условие выполнено.

$i$	$j$	$l$	$x$
-----	-----	-----	-----

Вместо блоков  $j, l, x$  будем подставлять  $i$ .

Для тех блоков, что останутся увеличиваем разбиение, и повторяем квантизацию с меньшим шагом. Практика показывает что для размеров блоков 2 на 2 пиксела всегда находится аналог.

Кодовые слова выбираются из так называемой кодовой книги или книги шифров. Размер кодовой книги влияет на скорость кодирования (число бит, необходимых для кодирования каждого вектора), а также на искажения. Чем больше кодовая книга тем выше скорость кодирования и меньше искажений. Для векторного квантования кодирование занимает больше ресурсов процессора чем декодирование. Кодирование требует поиска кодовой книги для каждого слова, в то время как декодирование требует только поиска по таблице. Обычно же кодовой книги используется кодер и декодер.

## Временное квантование

Для уменьшения временной избыточности в кодеке используется **кодирование разности** между кадрами. Нет никакого смысла кодировать два кадра отличающиеся друг от друга малым числом блоков. При этом вычитание кадров происходит по блокам, полученным ранее в результате векторной квантизации. Огромным преимуществом является то, что разницы между блоками являются малыми числами. Но используя кодирование разности, степень сжатия можно повысить. Для этого достаточно округлять разности до некоторой величины. Такое округление называется временным квантованием<sup>1</sup>. В данном случае квантование заключается в округлении вещественных чисел: математическом ( $0.6 = 1$ ) или финансовом ( $0.6 = 0$ ). При использовании такого округления со временем накапливается ошибка разности кадров. Для этого в Сіперак используются «ключевые кадры» — это кадры без сжатия. В рассматриваемом кодеке каждый пятнадцатый кадр является ключевым.

---

<sup>1</sup>В таком виде, оно используется и для сжатия звука



## 1.2.2. Indeo 5

Достоинства:

- нет блочных артефактов для относительно высокой степени сжатия;
- «оптимален» в отношении времени, степени сжатия, количества артефактов;
- проявление артефактов равномерно по всему изображению (потому они менее заметны);
- масштабируемость нагрузки;
- большое число дополнительных функций не связанных непосредственно со сжатием.

Недостатки:

- устарел и достаточно редок;
- требователен к ресурсам;
- нет быстрого доступа к произвольным кадрам при декомпрессии — это говорит о вычислительной сложности

Indeo 5 стал последним (во всех смыслах) кодеком этого семейства Indeo. Как было сказано на лекциях, кодек практически нигде не используется и является «мертвым». В доказательство последнего, можно сказать, что тестовые примеры с сайта Ligos Corporation (нынешней владелиц Indeo 5), под Windows Vista не запустились. Аналогичная ситуация наблюдается на Windows 7 и на Ubuntu 10.04<sup>2</sup>.

В кодеке Indeo 5 (4) используется является собственный алгоритм сжатия видео разработанной Intel — Indeo Video Interactive (IVI) Это совершенно другая технология по сравнению с предыдущими релизами Indeo, которые были основаны на векторной конкретизации как в Cinepak (Indeo 2 и Indeo 3). В отличие от большинства используемых на сей момент кодеков Indeo 5 основан не на дискретном косинус-преобразовании а на вейвлет-преобразовании. Кроме того IVI предлагает набор «интерактивных» возможностей, которые выходят за рамки традиционных услуг, предоставляемых кодеками. Они включают прозрачность (цветность-манипуляция), частичное декодирование, масштабируемость нагрузки на процессор и защиту паролем. Если рассматривать исключительно Indeo 5, то он имеет ряд несущественных отличий от Indeo 4. Хотя оба кодека используют IVI.

Интересно заметить, что в кодере FFDSHOW есть два разных кодека:

- Indeo 4,5 (*не 4.5, а именно 4, 5*);
- Indeo 5.

---

<sup>2</sup>Без установки дополнительных библиотек.

## Пространственное сжатие

Алгоритм IVI работает в цветовом пространстве YUV.

Где:

- $Y$  — яркость;
- $U$  и  $V$  — цвето-разностные компоненты, это хроматический синий и хроматический красный,

Для преобразования RGB в YUV используют следующие формулы:

$$\begin{aligned} Y &= 0,299 \cdot R + 0,587 \cdot G + 0,114 \cdot B & [0; 255] \\ U &= 0,564 \cdot (Y - B) & [-128; +128] \\ V &= 0,713 \cdot (Y - R) & [-128; +128] \end{aligned}$$

Формулы для обратного преобразования:

$$\begin{aligned} R &= Y + 1,14 \cdot V \\ G &= Y - 0,395 \cdot U - 0,581 \cdot V \\ B &= Y + 2,032 \cdot U \end{aligned}$$

Вейвлет-сжатие используется для преобразования изображения из пространственной области в частотную область. Изображение разлагается на квадратные блоки и выполняется вейвлет-преобразование с целью получения частот коэффициентов. После этого происходит квантование (округление). На этом этапе осуществляется самые большие потери. Далее полученные коэффициенты сжимаются методами без потерь качества (Хаффман).

Само по себе вейвлет-преобразование можно назвать расширенным преобразованием Фурье. Более подробно с вейвлетами можно ознакомиться в [2] и в [11].

Indeo 4 использует преобразование Хаара (для режима масштабируемости) и наклонное преобразование. В Последнюю версию Indeo 4 (только под Windows) используется дискретное косинус-преобразование.

Заметим, что любая функция  $\psi(t)$  может называться вейвлетом при соблюдении следующих условий:

- ее среднее значение, интеграл по всей прямой равен 0.
- функция быстро стремится к нулю на бесконечности  $\lim_{t \rightarrow \infty} \psi(t) = 0$ .

Indeo 5 использует только наклонное преобразование. Вейвлет Хаара в этом кодеке не используется<sup>3</sup>. Вместо него используется вейвлет Добеши (LeGall).

---

<sup>3</sup>Очевидно, из-за блочности, вызванной его структурой.

Для работы с видео используется так называемое дискретное вейвлет-преобразование (ДВП). ДВП-алгоритм основан на «пропускании» сигнала через фильтры низких и высоких частот. Подобные фильтры были реализованы задолго до появления цифровой техники. И можно с уверенностью сказать, что ДВП можно реализовать аппаратно. Низкочастотный фильтр выдает грубую форму исходного сигнала (на выходе получится расплывчатое, размытое изображение с низким разрешением). Высокочастотный фильтр выдает сигнал разности или дополнительной детализации. Выход фильтра высоких частот может быть разложен аналогично с изначальным сигналом. Таким образом может быть получена дополнительная детализация. Выходные значения называются коэффициентами преобразования.

Важно: сжатия здесь не достигается.

## Временное сжатие

Для компенсации временной избыточности используется GOP (Group of Pictures). Это группа изображений. Как правило она состоит из 12 кадров. Группа может состоять из кадров различных типов:

- 1) [I] — «интро» или опорный кадр, это базовое и самодостаточное при декодировании изображение. Степень сжатия минимальна.
- 2) [P] — (predicted) предсказанный, восстанавливается с участием предыдущего [P] или [I] кадра, кодирование осуществляется с компенсацией движения и предсказания вперед.
- 3) [B] — (bidirectionally predicted) двустороннее предсказывание. [B] кадр восстанавливается с участием предыдущего и/или последующего [P] или [I] кадра.

Порядок воспроизведения:

$[I_0][B_1][B_2][P_3][B_4][B_5][P_6][B_7][B_8][P_9][B_{10}][B_{11}]$

Порядок передачи и хранения:

$[I_0][P_3][B_1][B_2][P_6][B_4][B_5]...$

## Компенсация движения

Если кодер обнаружил, что часть [P] предыдущего кадра, как одно целое, переместилась в новое положение на текущем кадре, то [P] можно сжать, записав следующие три компонента:

- предыдущее местоположение;
- новое местоположение;
- информацию о границе области [P].

Компенсация движения будет эффективной, если объекты просто перемещаются по сцене, но не удаляются, приближаются или поворачиваются. Существенное изменение освещения сцены от кадра к кадру также снижает эффективность этого метода. Для текущего блока текущего кадра необходимо сделать поиск совпадающего или близкого к нему блока на предыдущем кадре. Но кроме того для поиска похожих блоков в кадрах используются различные методы.

### 1.2.3. MPEG-4 DivX

Достоинства:

- нет блочных артефактов степени сжатия (при соблюдении двухпроходности и для не высокой);
- наиболее современный и достаточно распространенный кодек;
- гибок — можно в широком диапазоне менять качество сжатого файла;
- большое число дополнительных функций не связанных непосредственно со сжатием.

Недостатки:

- требователен к ресурсам;
- в отличие от своего близнеца (XviD) — проприетарен;
- реализованы не все возможности стандарта;
- для получения сжатого видео хорошего качества нужна двухпроводность — сложно использовать в масштабе реального времени.

MPEG-4 на данный момент является последним из принятых стандартов линейки MPEG-N. MPEG-4 изначально был задуман как стандарт для работы со сверхнизкими потоками. Однако в процессе довольно долгой подготовки стандарт претерпел совершенно революционные изменения, и сейчас сжатие с низким потоком входит в него как одна составная часть, причем достаточно небольшая по размеру. Например, сам формат сегодня включает в себя такие вещи, как синтез речи, рендеринг изображений и описания параметров визуализации лица на стороне программы просмотра.

Разработка MPEG-7 была начата в 1996 г. К алгоритмам сжатия видео этот стандарт имеет еще меньшее отношение, чем MPEG-4, и не получил широкого распространение к данному моменту.

Стандарт MPEG, как и другие стандарты на сжатие, описывает лишь выходной битовый поток, неявно задавая алгоритмы кодирования и декодирования. При этом их реализация перекладывается на кодеки. Наиболее популярный, на сегодня, коммерческий кодек — DivX.

В некоторых кругах DivX справедливо называют «MP3 для видео».

Вообще говоря, DivX не реализует полностью стандарт MPEG-4, а реализует только его малую часть (MPEG-4 часть 2). Наряду с этой частью еще существует MPEG-4 часть 10, более известный как H.264.

Особенности:

- 1) Кодирование объектов имеющих не правильную форму.
- 2) Можно кодировать объекты переднего и заднего плана независимо.
- 3) Кодирование объектов анимации на основе двухмерных и трехмерных полигональных сеток (анимация лиц и фигур людей).
- 4) Масштабируемое кодирование, для гибкой схемы передачи на разных битовых скоростях.
- 5) Кодирование неподвижных текстур.
- 6) Поддержка профилей на уровне стандарта.

Так же в стандарт включены такие добавки как:

- 1) Расчет трехмерных сцен и работа с синтетическими объектами
- 2) Объектно-ориентированная работа с потоком данных.
- 3) Помещение в поток двоичного кода «C++ подобного» языка BIFS.
- 4) Активная зрительская позиция. Как было замечено выше, BIFS позволяет задавать реакцию объектов сцены на действия пользователя.
- 5) Синтезатор лиц и фигур.
- 6) Синтезатор звуков и речи.

Это весьма сомнительная функция на наш взгляд. Практическое ее применение, например для озвучивания или синхронного перевода фильма, даже в 2010 году невозможно.

В кодеке DivX, как можно судить эти возможности не реализованы.

## Пространственное сжатие

Для пространственного сжатия в данном случае используется алгоритм близкий к алгоритму JPEG. Изображение переводится из цветового пространства RGB в цветное пространство YUV. Исходное изображение разбивают на матрицы 8x8. Формируем из каждой три рабочие матрицы ДКП<sup>4</sup> — по 8 бит отдельно для каждой компоненты. При больших степенях сжатия этот шаг может выполняться чуть сложнее.

---

<sup>4</sup> Дискретное косинус-преобразование.

Изображение делится по компоненте Y, как и в первом случае, а для компонент U и V матрицы набираются через строчку и через столбец. Каждый блок подвергается ДКП:

$$C_{i,j} = \sum_{n=0}^{N-1} \sum_{m=0}^{M-1} C(i)C(j) \cos\left(\frac{(2n+1)i\pi}{2N}\right) \cos\left(\frac{(2n+1)j\pi}{2M}\right) f_{n,m}$$

$$C(i) = \begin{cases} \frac{1}{\sqrt{2}}, i = 0; \\ 1, i \neq 0 \end{cases}$$

При этом мы получаем матрицу, в которой коэффициенты в левом верхнем углу соответствуют низкочастотной составляющей изображения, а в правом нижнем — высокочастотной. Понятие частоты следует из рассмотрения изображения как двумерного сигнала (аналогично рассмотрению звука как сигнала). Плавное изменение цвета соответствует низкочастотной составляющей, а резкие скачки — высокочастотной. Полученную матрицу коэффициентов поэлементно делят на матрицу квантований.

$$A_{N \times N} = (a_{i,j})_{i,j=0}^{N-1}, \quad \text{где} \quad a_{i,j} = 1 + \alpha(i+j);$$

$\alpha$  — коэффициент качества, он задается пользователем. В случае MPEG-4,  $N = 8$ .

Для каждой компоненты (Y, и U и V) в общем случае задается своя матрица квантования. На этом шаге осуществляется управление степенью сжатия и происходят **самые большие потери**. После матрицу переводят в строку при помощи «зигзаг»-сканирования. Полученная последовательность сжимается по RLE. После того к последовательности применяется арифметическое кодирование.

Тут важно заметить, что согласно стандарту MPEG-4 (часть 2) ДКП может быть заменено на вейвлеты. Но используется это редко. В DivX оно используется только для статических изображений.

## Временное сжатие

В MPEG-4 так же используется метод компенсации движения. Об этом говорилось выше. Другой интересной особенностью MPEG-4 является компенсации глобального движения<sup>5</sup>. Но в данном кодеке метод реализован слабо и выгоды от его использования не много.

Последнее лучше проявило себя в XviD.

## Двухпроходность

Для того, чтобы получить качественное сжатое видео необходимо осуществлять два прохода кодека DivX. Вначале кодек строит таблицу перемещений, анализирует поток и записывает информацию о ключевых кадрах и уровнях квантизации всего потока. При втором потоке формируется сам сжатый файл. Есть крупное подозрение,

<sup>5</sup>GMC — Global motion compensation.

что на втором этапе кодек осуществляет постобработку сжатых кадров для уменьшения эффекта блочности. В итоге это приводит к увеличению сжатого файла.

Если использовать только один проход кодека, то сжатый файл будет иметь слишком большое количество артефактов<sup>6</sup>.

## 1.3. Объективные методы оценки качества

Существуют разные методы оценки качества. Бывают как оценки статичных изображений, так и оценки видео. Большая часть методов работает с яркостями (90 %). Остальные работают с оценкой цветности.

Все методы делятся на две большие группы:

- объективные, это повторяемые методы.
- субъективные или экспертные, предсказуемость этих методов страдает.

Одна из серьезных проблем машинной графики заключается в том, что до сих пор не найден адекватный критерий оценки потерь качества изображения.

Разработчикам методов сжатия изображений с частичной потерей информации необходимы стандартные метрики для измерения расхождения восстановленных изображений и исходных изображений. Чем ближе восстановленный образ к исходному, тем больше должна быть эта метрика (ее удобно называть «метрикой сходства»). Эта метрика должна быть безразмерной и не слишком чувствительной к малым изменениям восстанавливаемого изображения [9].

Объективные оценки являются с одной стороны более точными и правильными, но с другой их показатели тоже не являются идеальными в прикладном смысле.

Лучше всего потери качества изображений могут оценить эксперты.

В этой работе мы будем рассматривать методы:

- значение метрики  $\sup(x - y)$ , где  $x$  и  $y$  — пиксели исходного и сжатого кадров;
- значение математического ожидания разности кадров;
- значение метрики PSNR.

### 1.3.1. Метрика $\sup(x - y)$

Данная метрика вычисляет попиксельную разность двух изображений и находит максимум из этих значений. При всей своей простоте она крайне чувствительна к биению отдельных пикселей. То есть во всем изображении может существенно измениться только значение одного пикселя (что практически незаметно для глаза), однако согласно этой мере изображение будет сильно испорчено.

---

<sup>6</sup>Настолько большое, что исходного видео порой просто не видно.



Достоинства:

- простота понимания;
- простота реализации;
- можно использовать для нахождения максимальных различий;

Недостаток — чувствительность к «штучным» артефактам.

### 1.3.2. Математическое ожидание разности

Основная идея метрики — вычислить некоторое усредненное значение пикселей и сравнить их.

$$M = d(x, y) = \frac{\sum_{i=1, j=1}^{m, n} (x_{i,j} - y_{i,j})}{M \cdot N}, \quad M, N — \text{размерности}$$





На картинке приведено два изображения. Они «почти» не отличаются. Однако артефакт (который в этом случае введен искусственно) виден отчетливо. Однако метрика решит, что изображение изменилось не сильно.

Достоинства:

- не чувствительна к «штучным» артефактам.

Недостатки:

- не чувствительна к заметным артефактам;
- чувствительна к общему изменению яркости;
- не отражает субъективных оценок.

### 1.3.3. PSNR

Метрика которую сейчас часто используют на практике, называется мерой отношения сигнала к шуму (PSNR — Peak Signal to Noise Ratio):

$$\text{PSNR} = d(x, y) = 20 \log_{10} \frac{MAX \cdot M \cdot N}{\sqrt{\sum_{i=1, j=1}^{m, n} (x_{i,j} - y_{i,j})^2}}$$

- $MAX$  — пик сигнала, для полутоновых изображений это 255;
- $M, N$  — размерности.

Высокое значение PSNR означает определенную схожесть реконструированного и исходного изображений, но оно не дает гарантии того, что зрителю понравится восстановленный образ. Данная метрика фактически лишена тех минусов, которые были свойственны двум предыдущим. Но например следующие 2 изображения совершенно не различимы на глаз. А для метрики они будут совершенно различными, так как **второе изображение ярче**.



Правда, после появления артефактов типа «снег» или полос метрика покажет что изображения отличаются значительно. А подобные артефакты достаточно хорошо видны для глаза.

Достоинства:

- не чувствительна к «штучным» артефактам;
- чувствительна заметным артефактам.

Недостатки:

- чувствительна к общему изменению яркости;
- не отражает субъективных оценок.

Про эту метрику можно так же сказать, что изображения будут признаны значительно отличающимися если сжатое имеет множественные артефакты вне зоны внимания эксперта. Например, по краям. С формальной точки зрения и при детальном рассмотрении, изображения действительно будут сильно отличаться. При оценке, эксперты этот факт всего скорее не заметят. При неравномерной яркости (затухание к краям) ситуация будет аналогичной.

Мера PSNR по сути, аналогична среднеквадратичному отклонению, однако пользоваться ей несколько удобнее за счет логарифмического масштаба шкалы. Ей присущи те же недостатки, что и среднеквадратичному отклонению.

### 1.3.4. Контекстные методы

Так же еще существует большое число других методов объективной оценки качества. Их рассмотрение выходит за рамки данной работы, но мы тут их перечислим.

- 1) Just-noticeable difference (JND) — метод едва заметны различий.
- 2) Perceptual Distortion Metric (PDM) — метрика воспринимаемых искажений.
- 3) Structural similarity (SSIM) — метрика структурного сходства.

Эти метрики, пытаются лучше проанализировать отличия, но традиционно в большинстве случаев используется именно PSNR. Всего скорее это связано с тем, что реализация контекстных методов очень сложна сама по себе.

## 1.4. Субъективные методы оценки качества

Часто при оценке, вынуждены работать с субъективными методами, ибо «человек мера всех вещей».

Субъективные методы:

- шкала деградации с двумя стимулами<sup>7</sup>;
- непрерывная оценка качества с одним стимулом;
- попарное сравнение.

Требования к экспертам:

- размер группы от 4 до 40;
- *одинаковая* острота зрения и *нормальное* цветовосприятие;
- эксперты не должны быть профессионалами в области оценки качества.

### 1.4.1. Шкала деградации с двумя стимулами

Будем называть стимулом сам исходный видео-ролик. В этом методе участники просматривают эталонную сцену, а затем ту же сцену после ее обработки в тестируемой системе.

Каждый участник оценивает деградацию изображения сцены после обработки. одним числом по пятибалльной шкале:

- 1 — очень раздражает;
- 2 — раздражает;
- 3 — слегка раздражает;
- 4 — заметно не раздражает;
- 5 — не раздражает.

Баллы выставляются для некоторого количества различных сцен. Тут важно досматривать сцены до конца. И при показе нескольких вариантов одного и того же ролика — важно перед каждым показом сжатого файла показать оригинал.

---

<sup>7</sup>Double Stimulus Impairment Scale — DSIS.

### 1.4.2. Непрерывная оценка качества с одним стимулом

Зритель непрерывно оценивает одну телепрограмму длительностью 10-20 минут, а не серию тестовых сцен. Зрительные рейтинги записываются несколько раз в минуту. В этом методе зритель видит только сжатый ролик.

### 1.4.3. Попарное сравнение

На экране одновременно, два стимула. Пары создаются путем всевозможной комбинации из эталона и тестируемых методов.

Например, если  $A$  — исходный,  $B, C$  — тестируемый ролики, то пары будут:  $AB, BC, CB, BA, AC, CA$  и т.д.

При этом, эксперту не сообщается какой из роликов эталон. Пары должны размещаться на сером фоне (rgb: 0.5,0.5,0.5).

## 2. Вычислительная часть

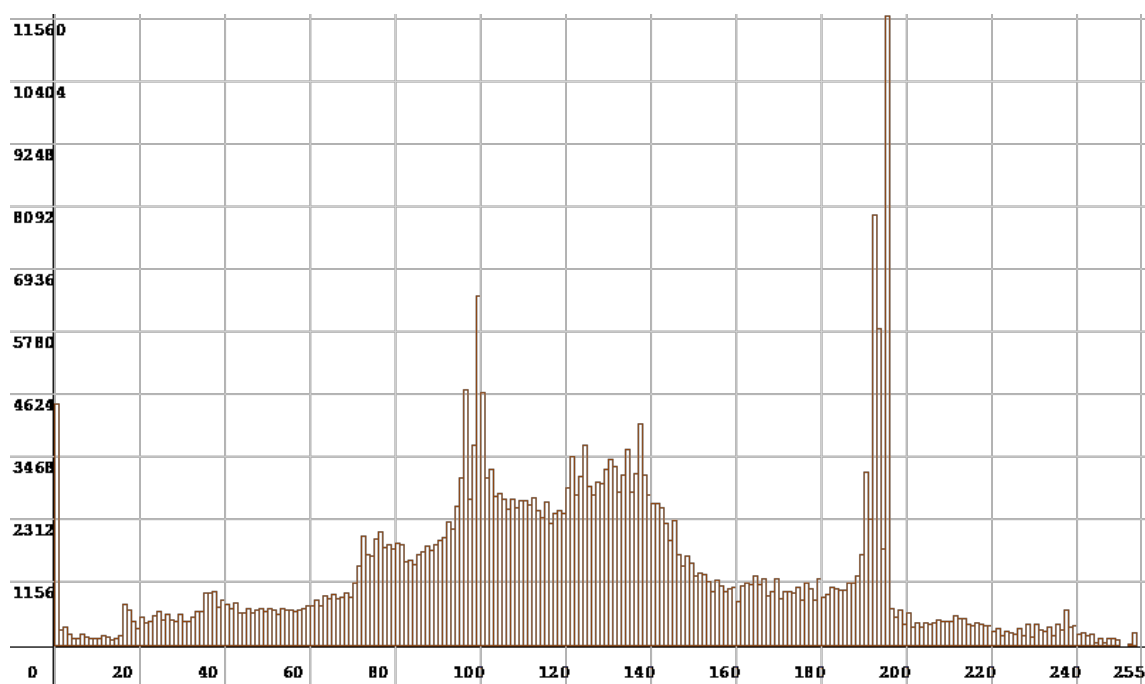
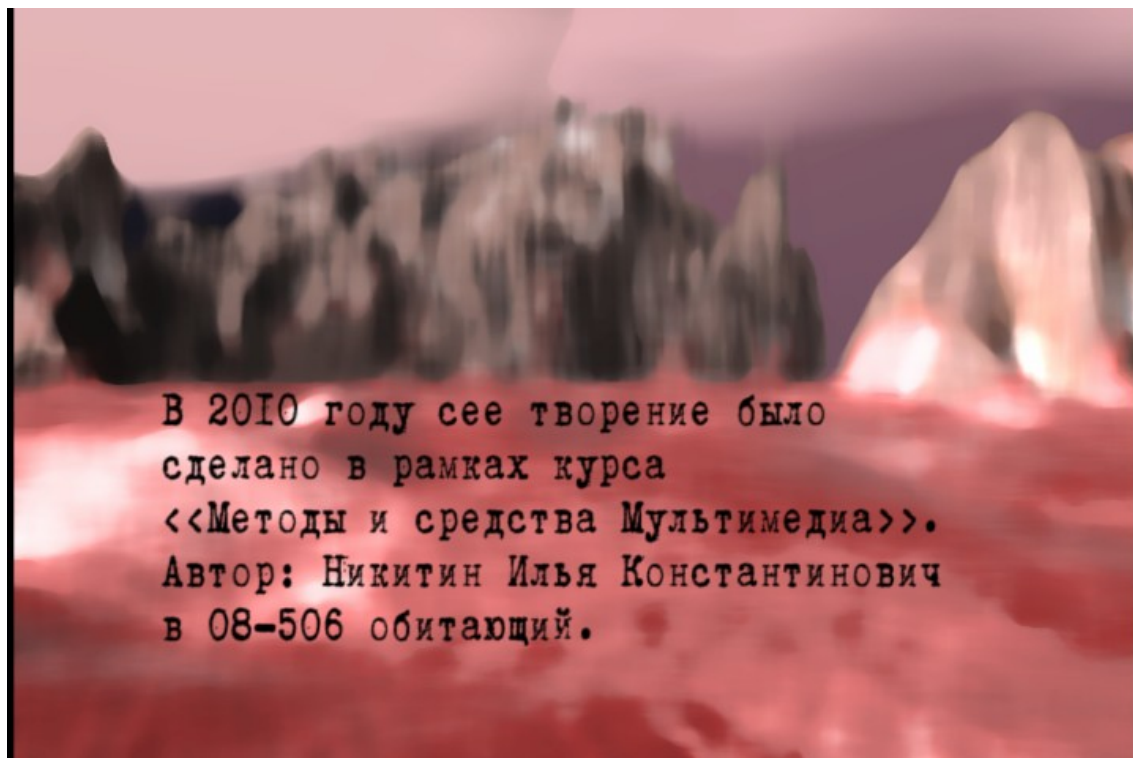
Первый кадр представляет собой титры на фоне гор, неба и «мяса» ... Второй — зависание около деревьев с монтированным видео. Третий кадр — фрагмент быстрого движения с мелкими деталями. Разрешение всего видео, и, соответственно, всех его кадров —  $720 \times 480$ . Всего видео насчитывает — 2046 кадров. Для объективной оценки качества использовалась программа Impartial Assessment, которая позволяет анализировать сжатые кадры:

- Supremum (супремум модулей разности яркостей соответствующих пикселей),
- PSNR (peak signal to noise ratio/пиковое отношение сигнала к шуму, измеряется в дБ),
- Average of distribution (математическое ожидание разностей).

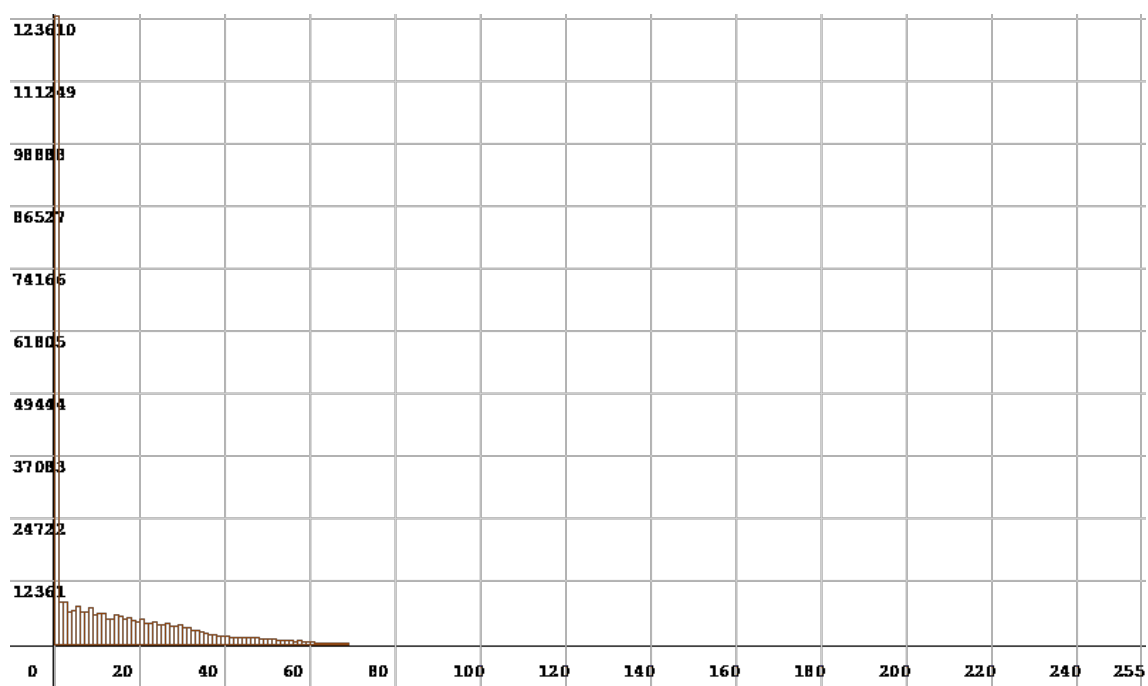
Далее приведены гистограммы яркостей кадров. Мы оставили здесь много свободного места, чтобы кадр и его гистограммы были на одной странице.

## 2.1. Гистограммы яркостей

### 2.1.1. Кадр 0055

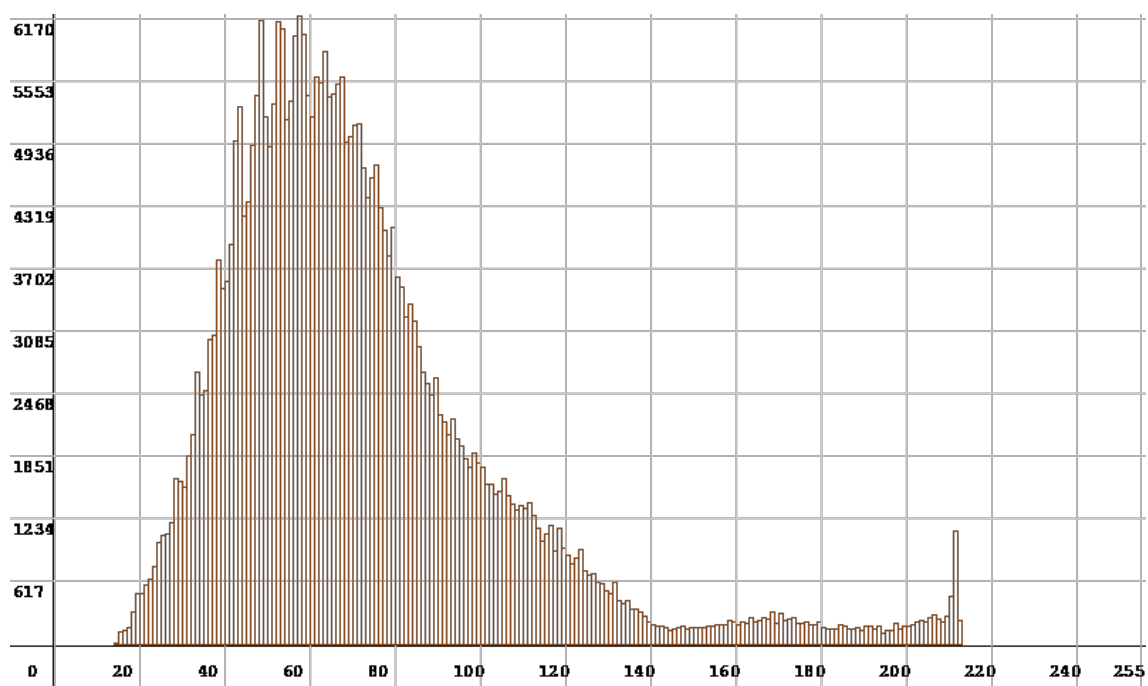
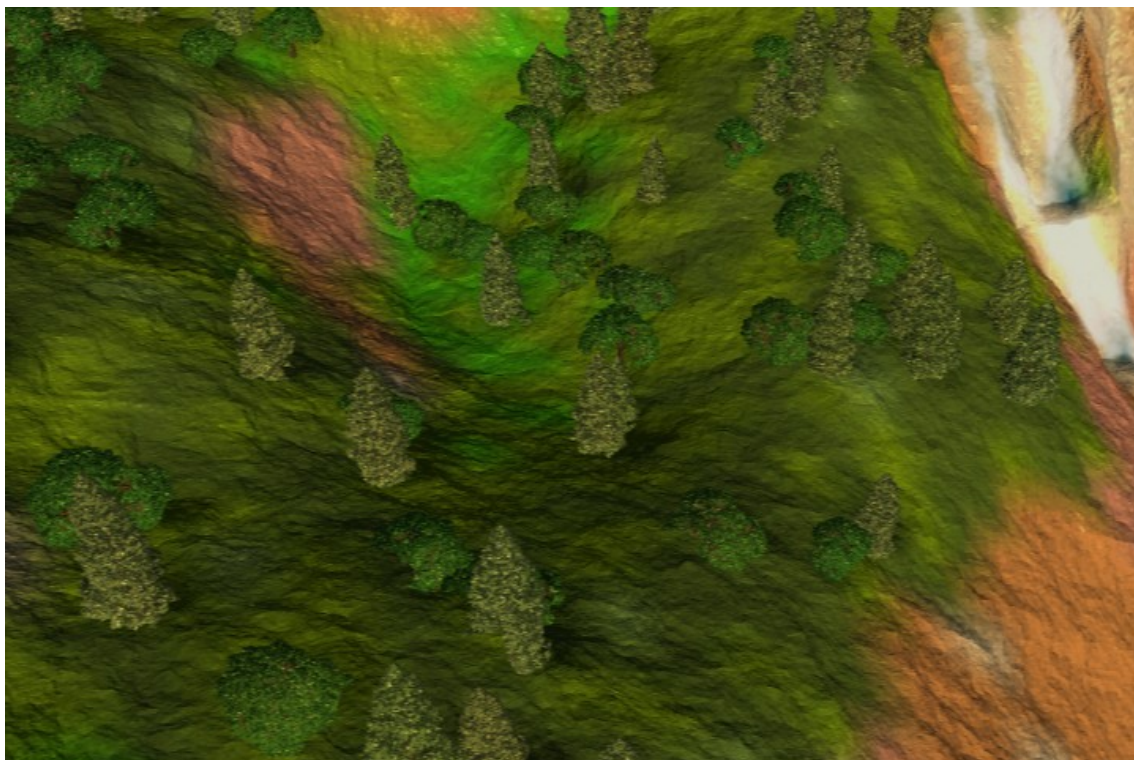


## 2.1.2. Кадр 0746





### 2.1.3. Кадр 1316



## 2.2. Субъективные оценки

Было проведено исследования по шкале деградации с двумя стимулами. В качестве испытуемых были выбраны 6 участников. Было проведено два этапа эксперимента. Эксперименты отличаются между собой степенью сжатия видео. Размер оригинала  $\approx 2$  Гб.

### 2.2.1. Низкая степень сжатия

Ниже приведена таблица для низких степеней сжатия. Из нее можно заключить, что в зависимости от кодеков мнения экспертов разнятся не сильно. В первой строке приведена степень сжатия в количестве раз по сравнению с оригиналом.

Испытуемые	Cinepak	Indeo5	DivX	Среднее
Размер	158 Мб	151 Мб	102 Мб	$\longleftrightarrow$
Сжатие (раз)	12	13	19	$\longleftrightarrow$
$L_1$	5	5	5	5
$L_2$	5	4	5	4.6
$C_1$	5	5	5	5
$C_2$	4	5	5	4.6
$C_3$	5	5	5	5
$C_4$	3	5	5	4.33
Среднее	4.5	4.83	5	$\longleftrightarrow$

DivX использовался двухпроходный.

Результаты этого тестирования нас не удовлетворили. Мы решили увеличить степень сжатия и провести еще одно исследование.

### 2.2.2. Высокая степень сжатия

Испытуемые	Cinepak	Indeo5	DivX	Среднее
Размер	158 Мб	43.5 Мб	38.3 Мб	$\longleftrightarrow$
Сжатие	12	45	52	$\longleftrightarrow$
$L_1$	4	3	2	3
$L_2$	4	3	2	3
$C_1$	5	4	3	4
$C_2$	4	4	3	3.6
$C_3$	5	4	3	4
$C_4$	3	3	3	3
Среднее	4.16	3.5	2.6	$\longleftrightarrow$

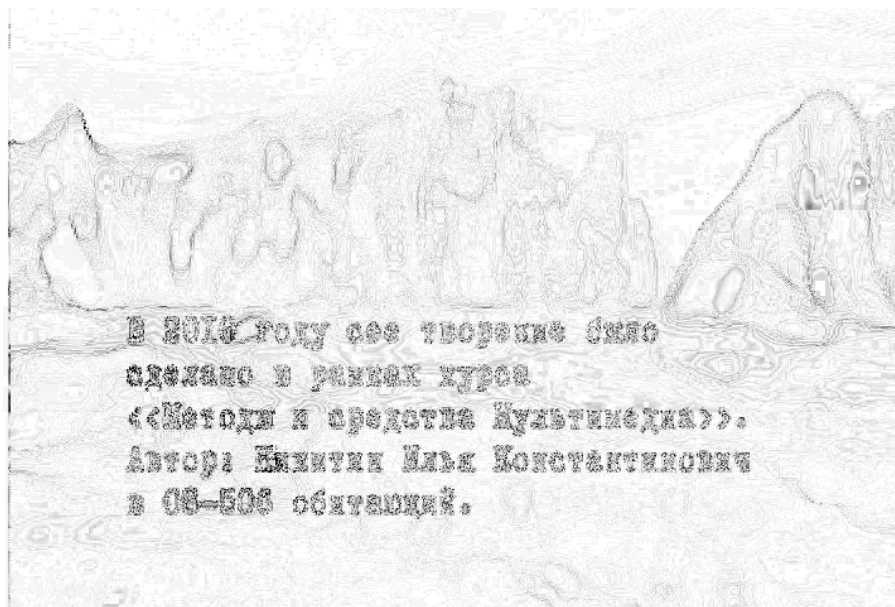
DivX использовался однопроходный  
с большим расстоянием между опорными кадрами.

Все последующие операции будут проводиться исключительно со результатами второго эксперимента.

## 2.3. Объективные оценки

### 2.3.1. Сінерак

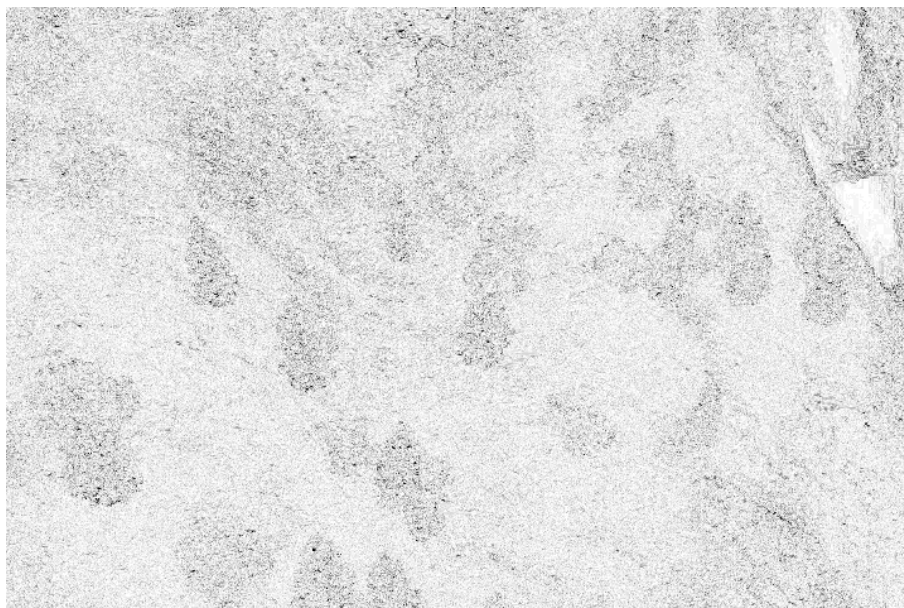
Кадр 0055 — линейная разность:



Кадр 0746 — линейная разность:



Кадр 1316 — линейная разность:



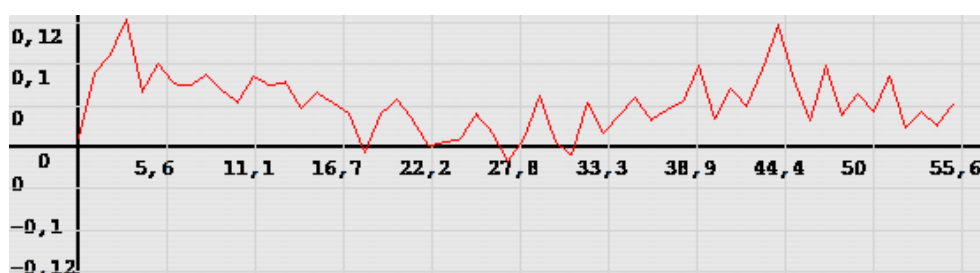
Кадр	$\sup(x - y)$	M	PSNR
0055	54	0.00863425925925926	37.3228044228198
0746	32	-0.234609375	40.0566925274575
1316	42	0.0629398148148148	36.022758718987

Графики для первых 55 кадров:

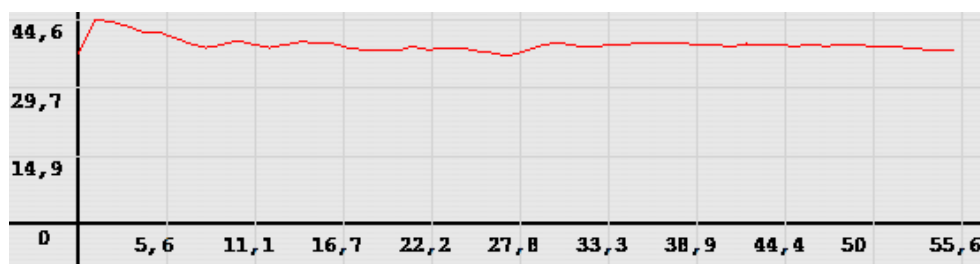
$\sup(x - y)$  :



Разность оценок матожидания :

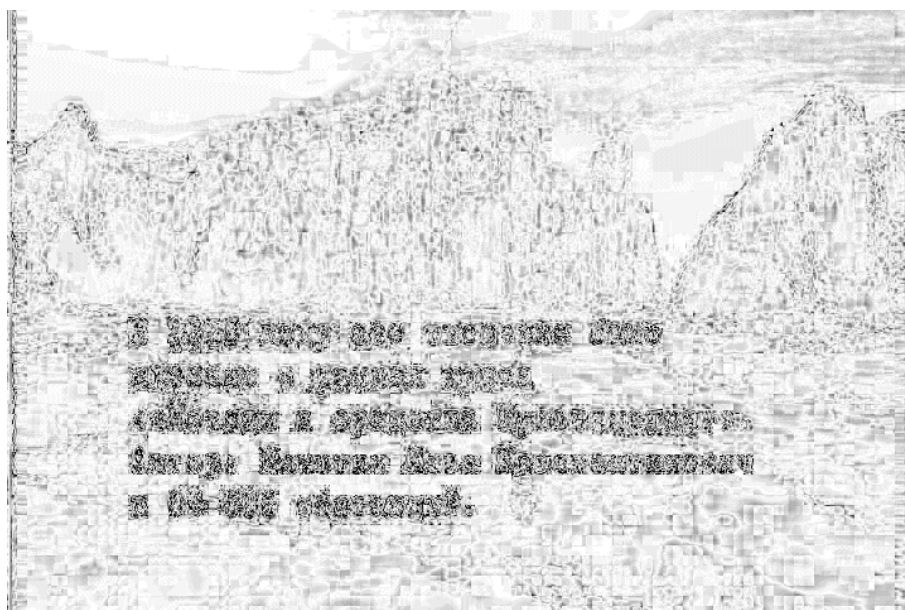


PSNR :



## 2.3.2. Indeo5

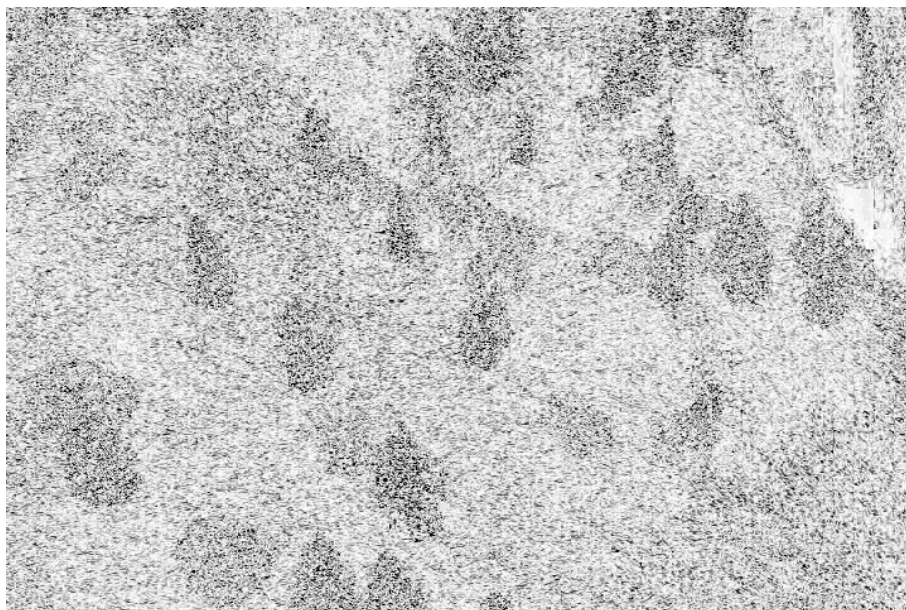
Кадр 0055 — линейная разность:



Кадр 0746 — линейная разность:



Кадр 1316 — линейная разность:



Кадр	$\sup(x - y)$	M	PSNR
0055	75	-0.415740740740741	33.1420663954279
0746	53	-0.878211805555556	33.7413058522762
1316	64	-0.303232060185185	30.4368685693292

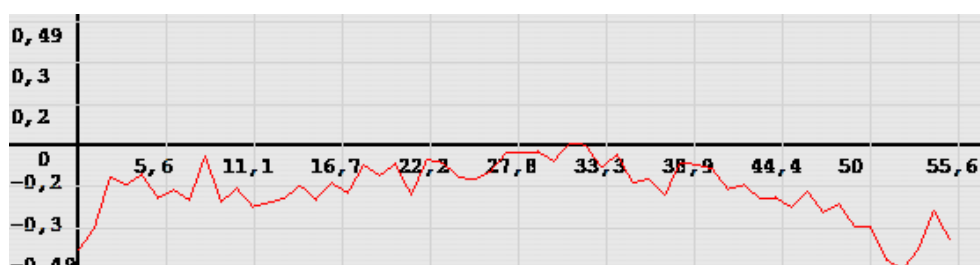


Графики для первых 55 кадров.

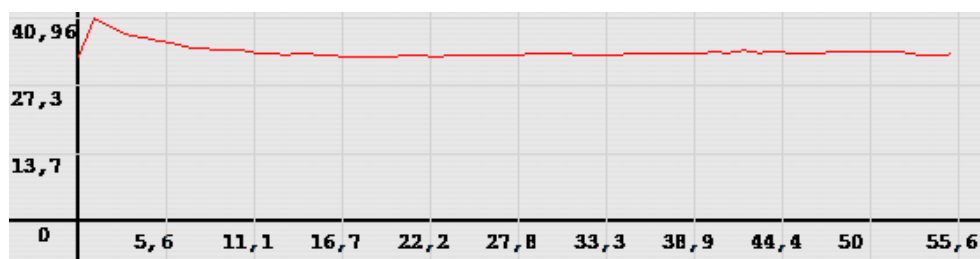
$\sup(x - y) :$



Разность оценок матожидания :



PSNR :





### 2.3.3. DivX

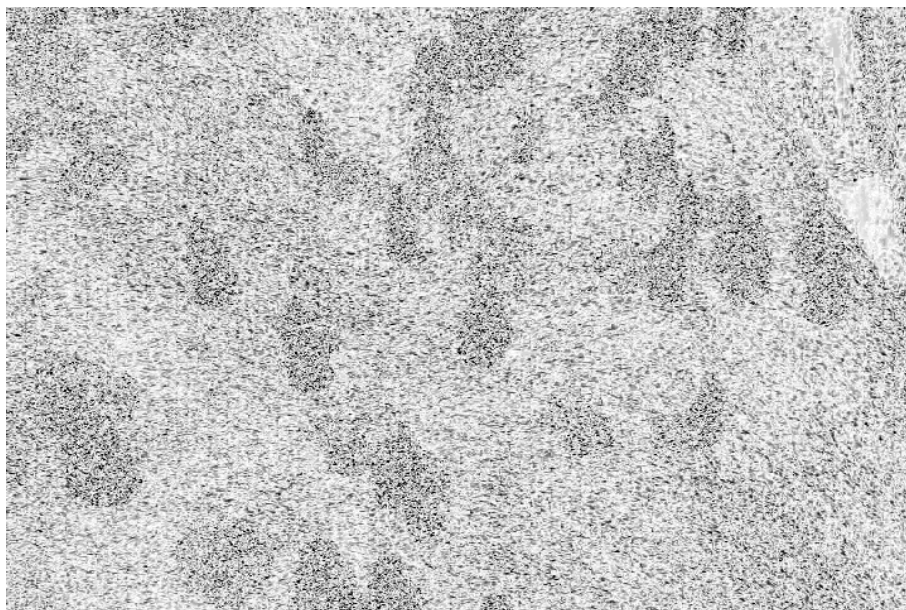
Кадр 0055 — линейная разность:



Кадр 0746 — линейная разность:



Кадр 1316 — линейная разность:



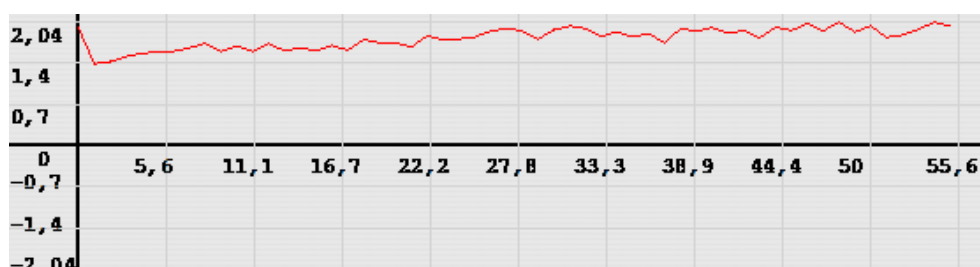
Кадр	$\sup(x - y)$	M	PSNR
0055	76	1.94266203703704	32.5942752164356
0746	36	0.212928240740741	36.3399041397657
1316	53	1.4008130787037	30.3073339805564

Графики для первых 55 кадров:

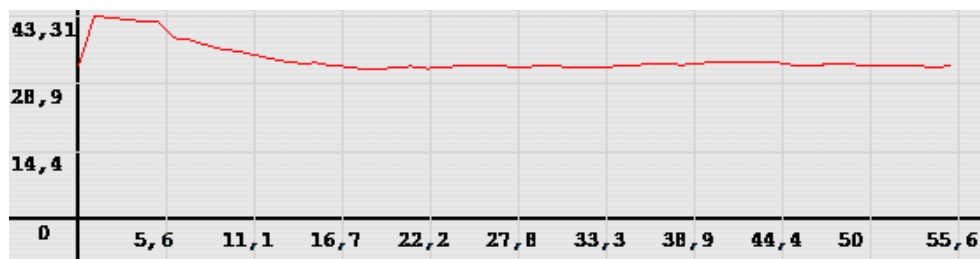
$\sup(x - y) :$



Разность оценок матожидания :



PSNR :



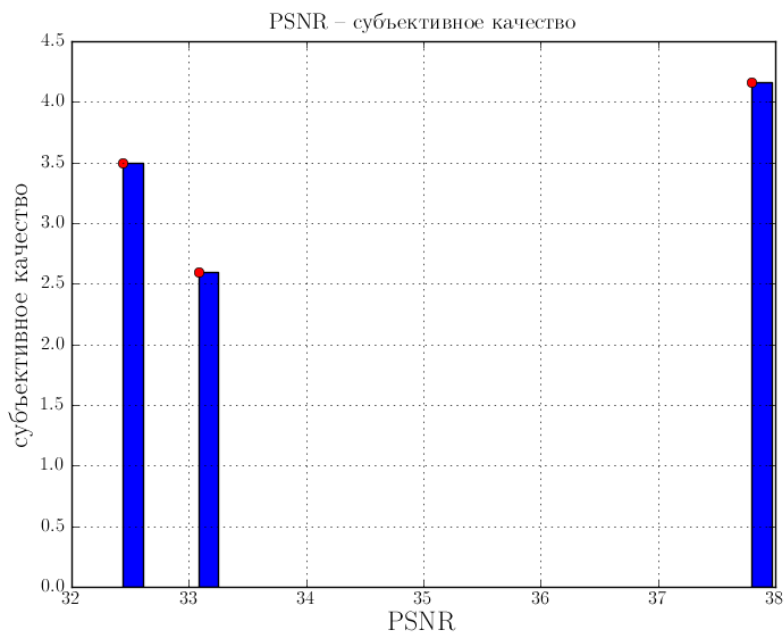
## 2.4. Сопоставление

### 2.4.1. Сопоставление оценок

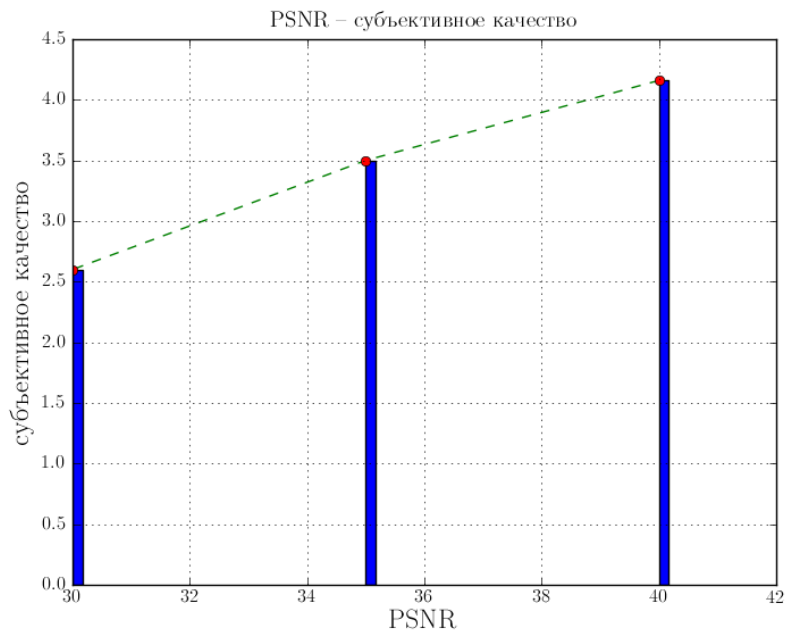
Если сопоставить объективные и субъективные методы оценки, то можно прийти к весьма противоречивым результатам. Мы знаем, как примерно должен выглядеть график сопоставления PSNR и субъективных оценок. Он должен придерживаться оценкам примерно следующим.

PSNR	Оценка
$\approx 20$	1
$\approx 25$	2
$\approx 30$	3
$\approx 35$	4
$\approx 45$	5

Причем, при  $\text{PSNR} = 45$  дБ, видео считается неотличимым от оригинала. Но проведя сопоставление только по трем тестовым кадрам мы получили:



Оценив («на глаз») PSNR для первых 55 кадров можно получить следующее.



Этот результат больше похож на правду. Всего скорее если провести сопоставление для всего видеоряда, то оно будет более соответствовать таблице выше.

«Лучшим» кодеком тут оказался Сінерак, а «худшим» — DivX. Что совпадает с субъективными оценками. Но еще раз обратите внимание, что Сінерак сжал исходный файл в 12 раз, а DivX в 52 раза.

Вообще говоря, такое расхождение для трех кадров можно объяснить причинами:

- малая выборка испытуемых и их индивидуальные особенности;
- всплески характеристик для отдельных кадров;
- неточностями (и ошибками) допущенными в ходе эксперимента.

Первое и последнее здесь мы рассматривать не будем. Хотя отрицать эти поправки нельзя.

Ниже приводится листинг программы с помощью которой мы строили графики.

---

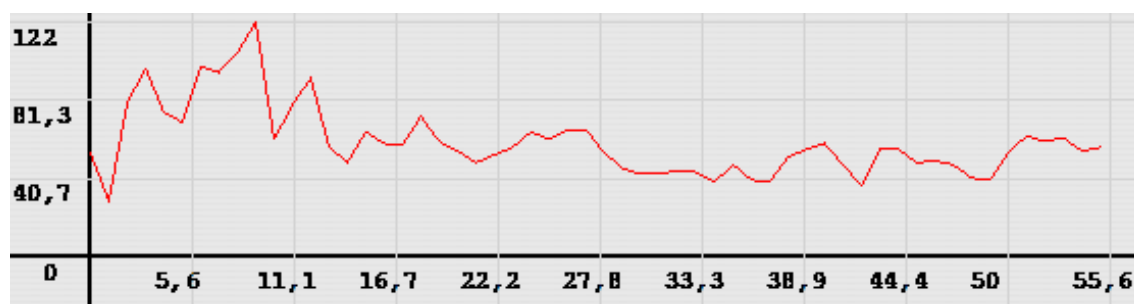
```
1  #! /usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4  import pylab
5  from config import *
6  def draw(points):
7      keys = points.keys()
8      values = points.values()
9      dim = len(keys)
10     w = 0.5
11     dimw = w / dim
12     f1 = pylab.figure()
13     pylab.plot(keys, values, 'g--')
14     pylab.bar(keys, values, dimw, bottom=0.001)
15     pylab.plot(keys, values, 'ro',)
16     pylab.grid()
17     pylab.legend(loc='best')
18     pylab.xlabel(u" PSNR", fontsize=16)
19     pylab.ylabel(ur" субъективное качество", fontsize=16)
20     pylab.title(u"PSNR - субъективное качество")
21     f1.savefig(GUI['picture-file'])
22     if(True == GUI['show']):
23         pylab.show()
24 if(__name__ == "__main__"):
25     impartial = {
26         'cinepak' : (37.3228044228198 + 40.0566925274575 + 36.022758718987) / 3,
27         'indeo5' : (33.1420663954279 + 33.7413058522762 + 30.4368685693292) / 3,
28         'divx' : (32.5942752164356 + 36.3399041397657 + 30.3073339805564) / 3
29     }
30     subjective = {
31         'cinepak' : 4.16,
32         'indeo5' : 3.5,
33         'divx' : 2.6
34     }
35     impartial_avg = {
36         'cinepak' : 40.0,
37         'indeo5' : 35.0,
38         'divx' : 30.0
39     }
40     draw({
41         impartial['cinepak'] : subjective['cinepak'],
42         impartial['indeo5'] : subjective['indeo5'],
43         impartial['divx'] : subjective['divx']
44     })
45     draw({
46         impartial_avg['cinepak'] : subjective['cinepak'],
47         impartial_avg['indeo5'] : subjective['indeo5'],
48         impartial_avg['divx'] : subjective['divx']
49     })
50 --
```

## 2.4.2. Сопоставление кодеков

Ниже еще раз приведены графики и таблицы для сопоставления по кодекам.

### sup (x-y)

Cinapak:



Indeo5:



DivX:

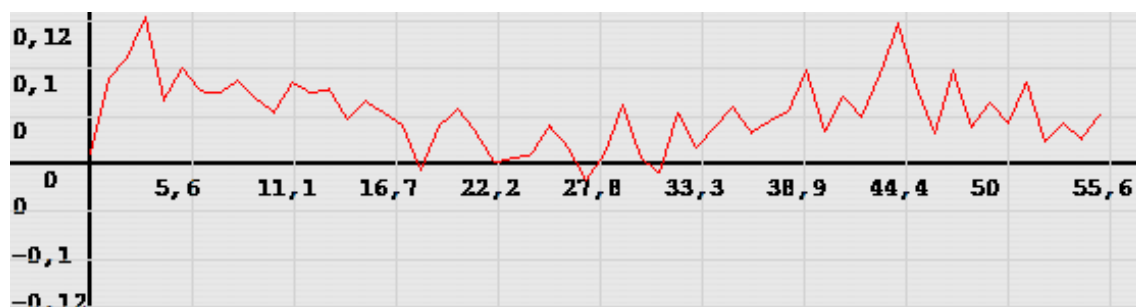


Таблица для трех отображенных кадров:

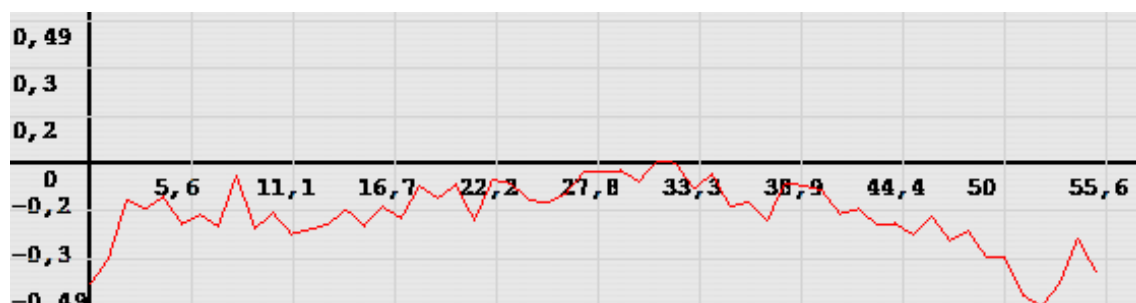
Кадр	Описание	Cinapak	Indeo5	DivX
0055	титры	54	75	76
0746	замедленный крупный план	32	53	36
1316	быстрое движение, мелкие детали	42	64	53

## Разность оценок матожидания

Cinapak:



Indeo5:



DivX:

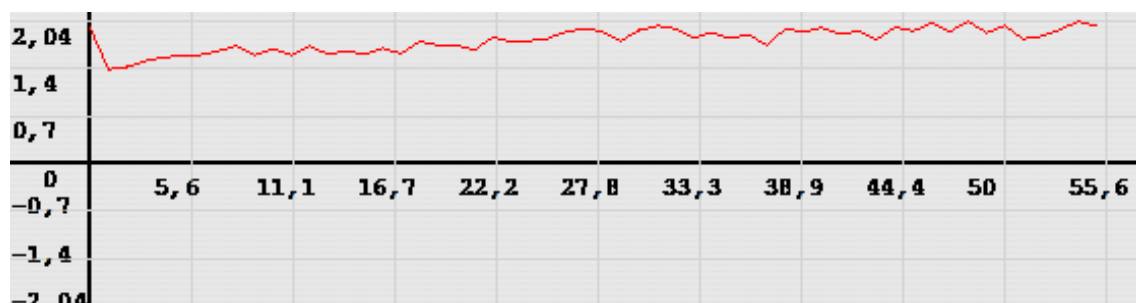


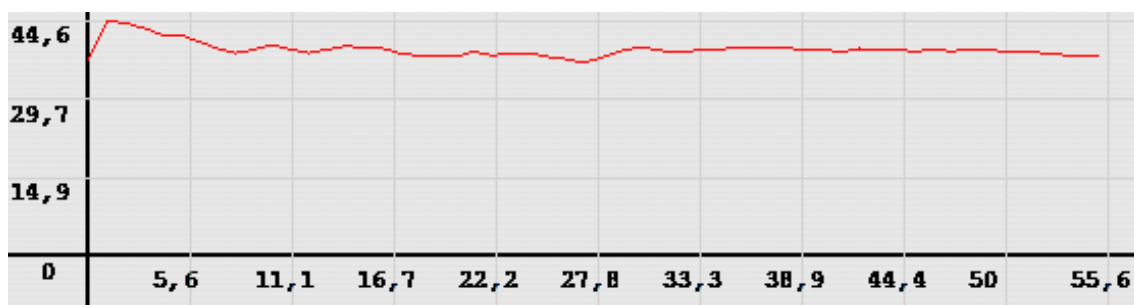
Таблица для трех отобранных кадров:

Кадр	Описание	Cinapak	Indeo5	DivX
0055	титры	0.0086	-0.4157	1.9426
0746	замедленный крупный план	-0.2346	-0.8782	0.2129
1316	быстрое движение, мелкие детали	0.0629	-0.3032	1.4008

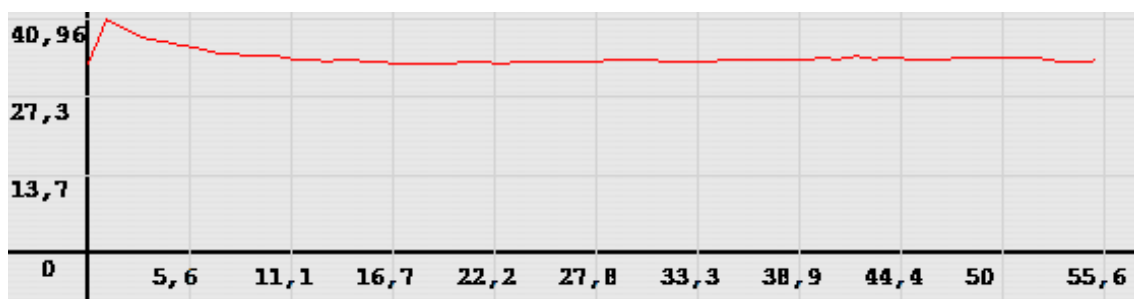


## PSNR

Cinepak:



Indeo5:



DivX:

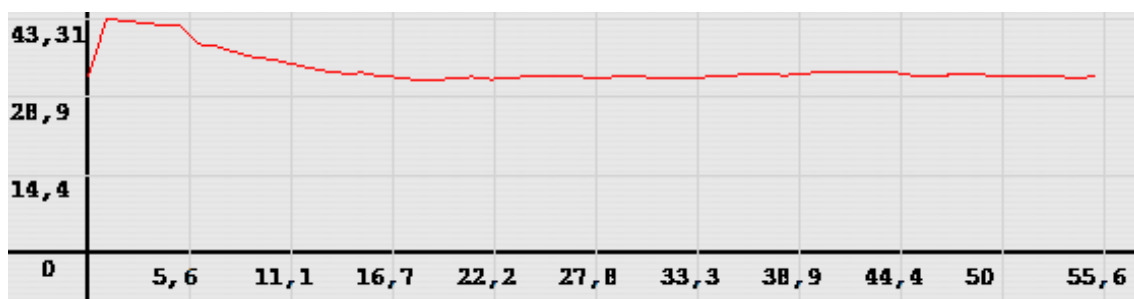


Таблица для трех отобранных кадров:

Кадр	Описание	Cinepak	Indeo5	DivX
0055	титры	37.32	33.14	32.59
0746	замедленный крупный план	40.05	33.74	36.33
1316	быстрое движение, мелкие детали	36.02	30.43	30.30

## 3. Аналитическая часть

Как было сказано выше, при сравнении субъективных оценок с PSNR, для выбранных трех кадров совпадения не наблюдается, но для рассмотренных первых 55 кадров, при некотором приближении, **можно говорить о совпадении**. В общем случае совпадения может не быть<sup>8</sup>. Проведем сопоставительный анализ результатов объективных и субъективных оценок по каждому кодексу.

### 3.1. Cinepak

В первую очередь испытуемые обращали внимание на шевелящуюся «сеточку» на границах с переходами цветов. Сеточка связана с векторной квантизацией используемой в кодексе. Шевеление — с временным квантованием. Только два эксперта заметили, что в Cinepak используется набор цветов, отличный от оригинального. Что касается объективных оценок, то действительно кодек хуже всего себя проявил на кадре с текстом. Текст на иллюстрации с иллюстрацией линейной разности виден отчетливо, это связано с блоками Cinepak. Границы блоков не обязательно должны попадать на границы объектов — здесь, текста, потому граница смазывается.

Теперь рассмотрим полученные значимые численные данные. При прочтении этого раздела рекомендуется иметь перед глазами графики из раздела «Объективные оценки».

#### 3.1.1. sup

$\sup(x - y)$  значительно изменяется, причем достаточно сложно предсказать его поведение. Это говорит о том, что имеется значительное число мелких артефактов предельная яркость которых меняется от кадра к кадру. Если судить о по первым 55 кадров, то метрика никогда не снижается до нуля и «скачет» в пределе [40, 122]. Всего скорее подобная картина наблюдается по всему ролику. Причина — блоки  $4 \times 4$  и временное квантование. На рассмотренных трех кадрах значения не являются максимальными их значения приведены в таблице.

#### 3.1.2. Разность оценок матожидания

Матожидание изменяется в малых пределах и это говорит, о том, что яркость кодек и меняет кардинальным образом. Малые колебания яркости может быть так же объяснена блочностью и округлением. По отрицательным значениям можно судить о том, что артефакты не только осветляют кадр, но и в некоторых случаях затемняет, но значения настолько малые, что всего скорее не надо на это обращать внимания.

---

<sup>8</sup>Причины этого сказаны выше. Если не очевидно, то есть две основные причины

- малая выборка испытуемых и их индивидуальные особенности;
- недостатки метрики PSNR, например чувствительность к общему изменению яркости.

### 3.1.3. PSNR

Если рассмотреть последовательность первых 55 кадров, то уровень держится в районе 40. В данном случае оценка весьма близка к субъективным, но в общем случае этого может не быть. Как мы писали выше, PSNR имеет ряд недостатков. При использовании этого кодека, вряд ли это недостатки смогут себя проявить, т.к. общая яркость в данном случае колеблется незначительно, а потери проявляют себя равномерно по всему кадру.

## 3.2. Indeo5

Для этого кодека было замечено легкое шевеление объектов в сценах с быстрым движением — «песочек». Один из экспертов охарактеризовал это как размытость сцены. В данном случае кадр 1316 как раз и является таким. На кадре много объектов, причем не очень много плавных переходов. Резкие тени камней и листья деревьев претерпели наибольшие изменения.

Чтобы судить о резкости переходов в этом кадре, совсем не обязательно видеть сам кадр. Достаточно, зная его *гистограмму яркостей*. Не трудно заметить, что на уровнях яркости [40, 80] есть значительно число неравномерных всплесков числа пикселей с одной яркостью, но число пикселей с близкими яркостями пилообразно изменяется. Это может говорить о большом числе относительно резких переходов. Строго судить нельзя, но можно выдвинуть предположение, которое в нашем случае подтверждается.

Это хорошо видно на линейной разности. Потому PSNR для этого кадра такой низкий. Если рассматривать этот кадр, то легко увидеть «лестничный эффект». Эффект также видно и для кадра 0746. Среди прочих кодеков Indeo5 испортил этот кадр сильнее. Тут не так много мелких деталей, но весьма много резких границ. Я думаю, в данном случае, это отчасти закономерно, а отчасти виновен и кадр. Нужно заметить, что учитывая, *гистограмму яркостей* этот кадр — очень темный (самые светлые пиксели находятся в районе 80 и их малое число). И то что он был сильно испорчен говорит, о том, что кодек осветляет. А как известно, такие перепады яркости являются слабым местом вейвлетов.

### 3.2.1. sup

$\text{sup}(x - y)$  значительно изменяется. Для первых 55 кадров он меняется в более широком диапазоне, но достаточно предсказуемо. На первых 12 кадрах всплеск связан с появлением титров и эффектами перехода. Они имеют резкие границы относительно остального фона. Имеет место проявление лестничного эффекта с не очень большим числом девиантных участков. Что таких участков не много можно судить о PSNR.

### 3.2.2. Разность оценок матожидания

Математическое ожидание в данном случае ниже нуля. Это означает, что данный кодек осветляет кадры на этом участки. Подобный эффект всего скорее может быть связан с частотным разложением и округлением значений, в результате чего некоторые темные детали теряются и потому происходит осветление. Отклонения очень незначительные на данном, и мы думаем, что на них можно не обращать особого внимания. Однако такого рода осветление для тесного кадра 0746 стало критическим.

### 3.2.3. PSNR

Как мы говорили ранее, график PSNR изменяется очень медленно и выглядит гладко. На 55 кадров уровень держится в районе 35 — нет сильных отклонений от субъективных оценок. График выглядит почти как прямая линия, за исключением всплеска в самом начале, а затем медленного спада. Всплеск говорит о том моменте, когда титры на экране еще не появились, а спад, о том что при появлении титров начал проявляться лестничный эффект. Однако его проявление, выглядит равномерным. Это говорит о том, что общее число артефактов и их усредненный вклад не сильно меняется от кадра к кадру.

## 3.3. DivX

Было замечено:

- 1) блочность — известный артефакт;
- 2) протяжка «хвостов» за движущимися объектами;
- 3) появление размытостей и мерцания.

Вообще как показала практика ролики сжатые DivX хуже масштабируются под размер экрана. И с увеличением области просмотра становится видно большее число артефактов. Но это мнение не экспертов а наше.

В данном случае при рассмотрении каждого кадра можно заметить блочность, Особенно это заметно опять на кадре 1316. Часть деталей оказалась просто будто бы замазана. Разницу сжатого и оригинального кадра хорошо видно по линейной разности. О том, что можно судить о детализации этого кадра по гистограмме яркостей, не имея самого кадра сказано выше.

Так же, этот кодек хуже остальных проявил себя на кадре 0055. На иллюстрации с линейной разностью отчетливо видны блоки. Далее в анализе будет показано, что DivX несколько затемняет изображения. И случай с кадром 0055 это подтверждает. Если опять обратиться к гистограмме яркостей, то можно сказать, что кадр достаточно ярко, причем крайне неравномерно. 7 % пикселей кадров<sup>9</sup> имеют уровень

---

<sup>9</sup>По гистограмме яркостей:  $\frac{8092 + 5780 + 11560}{720 \cdot 480} \cdot 100 = 7.3588$

яркости близкий к 200. Причем количество более ярких и более тусклых кадров значительно меньше.

Сжатый кадр в целом выглядит «ободранным» в некоторых местах пиксели даже явно поменяли свой цвет. Эффект рваности связан с характером алгоритма кодека. Сначала происходит дробление, а потом преобразование. С этим же связано, что некоторые деревья с кадра 1316 просто пропали.

### 3.3.1. sup

Изменение sup для этого кодека сильно отличается от всех прочих кодеков. С одной стороны, на первых 5 кадрах можно наблюдать значительный провал. Это говорит о том, что на этом участке нет сильных отклонений яркости отдельных пикселей. Подобное отчасти можно объяснить тем, что значительную часть кадра занимает темное небо и темные монотонные участки «поверхности». Несмотря на это артефакты имеют место, и об этом говорит не нулевые показания метрики. При появлении большей детализации в кадре значения начинают сильно меняться, Это говорит о том, что DivX добавляет очень большое число артефактов при появлении мелких деталей — в данном случае это блочность.

### 3.3.2. Разность оценок матожидания

Разность матожидания для этого кодека изменилась сильнее, чем для остальных. Причем в сторону уменьшения. Это говорит о том что, DivX затемняет кадры. Интересно заметить, что график растет — степень затемнения увеличивается. Для этого всего скорее нет общего, рационального объяснения и связано с появлением в кадре некоторого числа темных элементов, которые округляются до еще более темных, в конкретных блоках. Мы думаем, что рост графика является особенностью конкретно этого участка видео.

### 3.3.3. PSNR

На 55 кадров уровень держится в районе 30 — нет сильных отклонений от субъективных оценок. График имеет зазубрены, которые всего скорее связаны с [P] кадрами<sup>10</sup>. Это говорит, о том, что имеет место ошибка предсказания. Всплеск говорит о том моменте, когда титры на экране еще не появились, и большую часть кадра занимали монотонные темные объекты.

---

<sup>10</sup>Именно [P] кадры. [B] кадры при сжатии этого ролика не использовались.

## 3.4. Сопоставление кодеков друг с другом

### 3.4.1. sup

Если посмотреть данные трех тестовых кадров, то худшими кодеками по этой метрике оказались DivX и Indeo5. Высокие значения sup говорит о том, что имеется значительное число мелких артефактов — о так называемой «мусорности». Как мы говорили ранее, для кадров с резкими границами объектов Indeo5 проявил себя хуже. И наиболее чистым кодеком оказался Cinerak. Так как степень сжатия ролика была различной, мы не можем утверждать, что Cinerak на самом деле добавляет меньше мусора или он не так сильно разнится по яркости. Мы можем судить только, что это выполняется исключительно для выбранных кадров выбранных роликов. Если рассмотреть поведение метрики на первых 55 кадрах, можно заметить, что по абсолютному значению артефакты Indeo5 ярче, но среднее их значение на участке [22, 55] ниже чем у DivX. Сам график для Indeo5 стремится стать прямой. Это говорит, о том, что максимальная яркость артефактов этого кодека не сильно зависит от вида кадра, если там нет ярко выраженных границ<sup>11</sup>. Самым лучшим по абсолютному значению матрики на 55 кадров оказался Cinerak. Но поведение графика трудно предугадать — максимальная разница артефактов и оригинала от кадра к кадру может существенно измениться. Хотя артефакты в конкретном кадре могут быть не заметны, но их постоянное мигание при субъективной оценке замечено было. По данным графика можно сказать, что максимум достигается при появлении резких границ. Для DivX можно сказать, что мусорность тоже будет зависеть от конкретных кадров, о чем было сказано выше. Причем для темных кадров результат будет лучше.

### 3.4.2. Разность оценок матожидания

По отобранным трем кадрам, сложно сделать какие-то выводы о средней яркости. Можно выдвинуть предположение, что Cinerak и DivX затемняют, а Indeo5 осветляет. Но первых 55 это предположение подтверждается лишь отчасти. Действительно, Cinerak и DivX затемняют, а Indeo5 осветляет. Но на самом деле, на яркость кадра влияет только DivX, немного затемняя. Мы согласны, что его влияние на кадр очень не велико, но как минимум раз в 10 сильнее, чем влияние Cinerak и Indeo5. Тем более для DivX график имеет прослеживающуюся динамику роста (уменьшения средней яркости). Для Cinerak весьма сложно предугадать поведение графика, как и для метрики sup. Можно предположить, что оба графика ведут себя хаотически.

---

<sup>11</sup>Почему, смотрите выше.

### 3.4.3. PSNR

Внешний вид графиков для 55 кадров PSNR не отличается кардинальным образом. Отличаются только абсолютные значения для конкретных кадров. И все же график для Сінерак имеет более заметные колебания чем остальные, что «шумы» в кодированном ролике появляются в случайном порядке. Для Indeo5 график отличается своей стабильностью, что говорит о равномерном проявлении артефактов. Для DivX график на [17, 55] кадрах ведет себя тоже относительно стабильно. Причина всплесков в начале ролика объяснялась ранее. По графикам PSNR, можно заключить, что «Лучшим» кодеком тут оказался Сінерак, а «худшим» — DivX. Если рассматривать, только отобранные три кадра, то такого совпадения не наблюдается. При аналитическом разборе кодеков было указан ряд причин, по котором тот или иной кодек показал себя хуже на конкретном кадре.

# Заключение

Проведя это исследование, можно с уверенностью утверждать, что лучше не сравнивать кодеки на произвольном наборе изображений. Каждый кодек, как и метод, имеет свои преимущества. Например, очень не умно использовать вейвлеты для резких контрастных изображений или ДКП для плавных переходов. Отчасти по этой причине в стандарте MPEG 4 часть 2 было предусмотрено использование вейвлетов для статических изображений. С другой стороны, при близких степенях сжатия кодеки имеют очень разнящийся размер файла, и время компрессии/декомпрессии. Это нужно учитывать, при выборе кодека под конкретную задачу. Для оценки сжатых видео удобно использовать объективные оценки качества. Но использовать их надо аккуратно, потому что смотреть сжатые и испорченные ролики придется не машине, а человеку. И только на основе многочисленных экспертных оценок можно делать выводы о приемлимости того или иного варианта. К сожалению, многочисленные экспертные оценки — скорее утопия для реальных задач. Потому приходится довольствоваться PSNR, или контекстными методами. PSNR используется чаще. И что бы понять это достаточно прочитать пару тем на специализированных форумах. В данном случае мы используем PSNR, а не СКО, потому, что данная мера несколько удобнее за счет логарифмического масштаба шкалы. С другой стороны PSNR показывает не сколько мы потеряли, а сколько «качества» у нас осталось. Благодаря этим двум поправка удобно сопоставлять (хотя бы приблизительно) субъективные и объективные оценки качества.

В результате проведенного исследования, было так же наглядно продемонстрировано, что для анализа потерь качества, не обязательно видеть исходный и сжатый файл. Вполне достаточно иметь:

- гистограммы яркости — для анализа яркости и ее распределения.
- значения мертики  $\sup(x - y)$  — для поиска т.н. «мусора» — сильно выдающихся артефактов;
- значение разности оценок матожидания — для определения динамики яркостной характеристики;
- значение мертики PSNR.

На основе разности оценок матожидания можно корректировать оценки полученные в результате PSNR. И это компенсирует ее недостаток.

Мы признаем, что субъективные оценки намного лучше объективных, хотя бы потому, что видео «тестируется» в реальных условиях, для которых создавалось. Но субъективные оценки невозможно повторить, а для получения точных данных нужно иметь большую выборку (например, человек 40 или более). Субъективные оценки лишены этих недостатков, и после дополнительного анализа могут послужить основой для выводов о применимости того или иного метода сжатия.



# Список литературы

- [1] Bock A M., «Video Compression Systems: From first principles to concatenated codecs» — London: The Institution of Engineering and Technology, 2009 — 300 с.
- [2] Добеши И. «Десять лекций по вейвлетам»  
— И.: НИЦ «Хаотическая и регулярная динамика», 2001 — 464 с.
- [3] ISO IEC 14496 2 (mpeg 4 part 2) «Information technology — Coding of audio-visual objects» — ISO/IEC, 2001 — 536 с.
- [4] Ghanbari Mohammed, «Standard Codecs: Image Compression to Advanced Video Coding» — Institution of Electrical Engineers, 2003 — 407 с.
- [5] Dzung Tien Hoang, Jeffrey Scott Vitter «Efficient algorithms for MPEG video compression» — Wiley-Interscience, 2002. — 169 с.
- [6] Richardson I.E.G., «Video Codec Design»  
— John Wiley & Sons, 2002 — 304 с.
- [7] Richardson I.E.G., «H.264 and MPEG-4 Video Compression»  
— John Wiley & Sons, 2003 — 306 с.
- [8] Крапивенко А.В. «Технологии мультимедиа и восприятие ощущений: учебное пособие.» — М. : БИНОМ. Лаборатория знаний, 2009. — 271 с.
- [9] Сэломон Д. «Сжатие данных, изображений и звука»  
— М.: Техносфера, 2004. — 368с.
- [10] Ватолин Д., Ратушняк А., Смирнов М., Юкин В. «Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео».  
— М.: ДИАЛОГ-МИФИ, 2003. — 384 с.
- [11] Уэлстид С. «Фракталы и вейвлеты для сжатия изображений в действии.»  
(Учебное пособ.) — М.: Издательство Триумф, 2003 — 320 с.
- [12] Wootton C., «A Practical Guide to Video and Audio Compression»  
— APR, 2005 — 800 с.
- [13] Yun Q. Shi, Huifang Sun, «Image and Video Compression for Multimedia Engineering: Fundamentals, Algorithms, and Standards (Image Processing Series)»  
— CRC Press LLC, 200 — 455 с.
- [14] Материалы сайта [multimedia.cx](http://multimedia.cx).
- [15] Материалы сайта [compression.ru](http://compression.ru).
- [16] Материалы сайта [siggraph.org](http://siggraph.org).