

КОМПРЕССИЯ ULTRA HD-ВИДЕОИНФОРМАЦИИ В РАСПРЕДЕЛЕННОЙ КОМПЬЮТЕРНОЙ СИСТЕМЕ

Рассмотрены преимущества и недостатки современного формата представления видеоинформации Ultra HD. Выполнена оценка вычислительной сложности операции поиска векторов движения в видеопоследовательности. Проанализирован метод распределения видеоинформации по узлам распределенной системы, который использован для распараллеливания процесса компрессии видеоинформации на кластере. Экспериментально исследованы объективные показатели результата компрессии Ultra HD-видеоинформации в распределенной системе в зависимости от применяемого алгоритма оценки движения и ширины области поиска векторов движения.

Ключевые слова: Ultra HD-видеоинформация, распределенная компьютерная система, компрессия, битрейт, коэффициент сжатия, уровень искажения, корреляция.

ВВЕДЕНИЕ

Активное развитие формата телевидения высокой четкости (HD television) по всему миру способствовало значительному повышению качества представления видеоконтента. В 2013 году на международной конференции Consumer Electronics Show был представлен формат Ultra HD, суть которого заключается в повышении разрешения видеоконтента с 1920x1080 пикселей до 3840x2160 (4K UHD) и 7680x4320 (8K UHD), максимальная частота кадров повышена с 50 до 120 прогрессивных кадров в секунду [1]. Помимо положительных моментов, новый формат телевидения вносит целый ряд проблем: значительное увеличение объемов хранимых данных, повышение требований к производительности компьютерных систем, осуществляющих обработку и компрессию такой видеоинформации, ужесточение требований к каналам передачи данных.

Одним из путей решения таких проблем может быть применение параллельных и распределенных компьютерных систем для компрессии Ultra HD-видеоинформации. Это позволит добиться ускорения вычислительного процесса и снизить время сжатия видеоконтента.

ПОСТАНОВКА ЗАДАЧИ

Рассмотрим основную операцию любого современного алгоритма сжатия видеопоследовательностей, обладающую наибольшей вычислительной сложностью во всей структуре сжатия, – операцию оценки движения (ОД) блоков кадров видеопоследовательности [2]. Ее суть заключается в следующем: кадры разбиваются на блоки, далее выполняется ОД, при которой в текущем кадре, для каждого блока осуществляется поиск в предыдущем кадре наиболее «похожего» блока, смещенного на вектор, называемый вектором движения.

Таким образом, для каждого блока C_n в текущем кадре с номером n выполняется поиск блока C_k в ограниченной области предыдущего кадра с номером $k < n$, соответствующий минимальному значению некоторо-

го критерия. В качестве такого критерия чаще всего применяется сумма абсолютных различий значений пикселей блоков ($SAD \rightarrow \min$):

$$SAD(v) = \sum_{(x,y) \in C_n} |C_n(x,y) - C_k(x+v_x, y+v_y)|, \quad (1)$$

где $v = (v_x, v_y)$ – вектор движения; $|v_x| \leq \Omega$ и $|v_y| \leq \Omega$, Ω – ширина области поиска; $C_n(x, y)$ – значения яркостных и цветоразностных компонент пиксела с координатами (x, y) в блоке C_n ; $C_k(x+v_x, y+v_y)$ – значения яркостных и цветоразностных компонент пиксела с координатами $(x+v_x, y+v_y)$ в блоке C_k , смещенном на вектор движения v .

Найденные вектора движения кадра используются при его кодировании, что позволяет сократить временную или интеркадровую избыточность видеопоследовательности.

Выполним оценку вычислительной сложности алгоритма полного перебора векторов движения видеопоследовательности. Данный алгоритм позволяет получить наилучшее качество результата сжатия при наибольшей вычислительной сложности. Его вычислительную сложность будем использовать для оценки наихудшего случая времени сжатия видеопоследовательности. Пусть видеопоследовательность состоит из N кадров разрешением $W \times H$ пикселей каждый. Алгоритм оценки движения делит кадр на блоки размером $a_{size} \times b_{size}$ пикселей. Поиск векторов движения выполняется в области шириной Ω пикселей. Тогда оценка вычислительной сложности ОД видеопоследовательности в общем случае для всех цветовых моделей может быть представлена в виде следующего выражения:

$$O \left[(N-1) \cdot \frac{W \cdot H}{a_{size} \cdot b_{size}} \cdot (2 \cdot \Omega + 1)^2 \right]. \quad (2)$$

Из этой формулы несложно заметить, что увеличение разрешения кадра видеопоследовательности в формате Ultra HD приведет к линейному повышению времени выполнения ОД для сжатия. Алгоритмы так называемой «быстрой» оценки движения выполняют ОД в ограниченной некоторым шаблоном (например, крест, шестиугольник и т. д.) области поиска [2]. Это позволяет существенно уменьшить величину Ω и, тем самым, ценой снижения качества результата сократить время сжатия видеопоследовательности.

Целью данной работы является распараллеливание процесса компрессии Ultra HD-видеоинформации в распределенной системе (секции 1 и 2), получение объективных показателей результата компрессии при различных параметрах кодера (секция 3), а так же оценка современного стандарта сжатия видеопоследовательностей H.264 для такой видеоинформации.

1 РАСПАРАЛЛЕЛИВАНИЕ ПРОЦЕССА КОМПРЕССИИ ULTRA HD-ВИДЕОИНФОРМАЦИИ

Для распараллеливания компрессии видеоинформации в распределенной системе существуют несколько подходов, которые принципиально различаются зернистостью параллелизма. К наиболее часто применяемым подходам можно отнести следующие [3–5]:

1. Деление кадра на блоки и независимое сжатие блоков на узлах системы. Такой подход позволяет устранять только пространственную или интракадровую избыточность, поскольку для устранения временной избыточности необходимы минимум два смежных кадра. Видеоинформация в формате Ultra HD обладает более высоким разрешением кадра по сравнению с HD, вследствие чего, число образуемых блоков будет велико (например, 8100 блоков размером 64x64 пиксела) и зернистость параллелизма будет слишком мелкая. Другим недостатком такого подхода является увеличение накладных затрат: транспортировка по сети и объединение большого числа блоков от различных узлов в кадры.

2. Независимое сжатие отдельных кадров на узлах системы. Этот подход так же позволяет устранять только пространственную избыточность, однако зернистость параллелизма крупнее, чем в предыдущем подходе и накладные затраты меньше. Параллельный алгоритм MJPEG использует этот подход для сжатия видеоинформации.

3. Независимое сжатие отдельных групп смежных кадров. Подход позволяет параллельно устранять как временную, так и пространственную избыточность видеоинформации. Однако, результирующий средний битрейт и коэффициент сжатия зависят от конкретных мест деления видеопоследовательности на группы кадров. Зернистость параллелизма самая крупная из рассмотренных подходов, накладные затраты самые маленькие.

В результате анализа наиболее часто применяемых подходов к распараллеливанию компрессии видеоинформации заметно, что самым перспективным подходом к распараллеливанию сжатия видеоинформации в формате Ultra HD является независимое сжатие групп смеж-

ных кадров. Для эффективной компрессии такой видеоинформации необходимо устранение и пространственной и временной избыточности.

Применение графического процессора (GPU) для распараллеливания процесса компрессии видеоинформации позволяет получить существенное ускорение по сравнению с центральным процессором (CPU), однако это приводит к снижению качества результата сжатия до 10 % [6].

Рассмотрим предложенный в [7] метод распределения видеоинформации по узлам распределенной системы. Он основан на корреляционном анализе, с динамической чувствительностью анализатора, смежных кадров видеопоследовательности, рассматриваемых в качестве временных рядов. Суть метода заключается в том, что для каждой пары смежных кадров вычисляется коэффициент корреляции $r_{YUV}^{k,k+1}$ (k – номер кадра), учитывающий яркостную и цветоразностные компоненты пикселей, а так же вычисляется значение функции чувствительности к смене сцен анализатора видеопоследовательности $\varphi(k)$. Скорость роста функции чувствительности управляется параметром. Далее сравнивается модуль коэффициента корреляции для данной пары смежных кадров и значение функции чувствительности. Если $|r_{YUV}^{k,k+1}| > \varphi(k)$, тогда анализ видеопоследовательности продолжается: вычисляются $r_{YUV}^{k,k+1}$ и $\varphi(k)$ для следующей пары кадров и снова выполняется сравнение. Иначе, видеопоследовательность разделяется на две части: первая часть содержит кадры анализируемой видеопоследовательности до места, в котором $|r_{YUV}^{k,k+1}| \leq \varphi(k)$, и она отправляется на узлы системы для выполнения сжатия, вторая часть содержит кадры после данного места и ее анализ продолжается.

Преимуществом данного метода является нивелирование роста битрейта при сжатии видеопоследовательности в распределенной системе при сохранении того же уровня искажения, что и при последовательной компрессии. То есть коэффициент сжатия видеопоследовательности не ухудшается вследствие распараллеливания. Такие видеопоследовательности после разделения ограничены одной сценой и их смежные кадры характеризуются значительной корреляционной зависимостью, благодаря чему ОД может быть выполнена на относительно небольшой области поиска Ω . Недостатком метода является рост накладных расходов. Так, при сжатии видеоинформации в формате HD 1920x1080 по стандарту MPEG4/H.264 накладные расходы применения метода составляют 7,83 % от общего времени вычислительного процесса [7].

Для распределения Ultra HD-видеоинформации по узлам компьютерной системы будем использовать рассмотренный метод.

2 ОРГАНИЗАЦИЯ ЭКСПЕРИМЕНТОВ

Целью экспериментов является установление объективных показателей результата компрессии видеоинфор-

мації, а так же показателів ефективності застосування розподіленої системи:

- час стиснення, витрачений на процес компресії відеопослідовальності;
- рівень искаження стисненої відеопослідовальності відносно вихідної, який характеризує якість стиснення. В даній роботі він вимірюється в дБ за допомогою метрики PSNR [8];
- коефіцієнт стиснення відеопослідовальності – відношення розміру файлу, що містить стиснену відеопослідовальність, до розміру файлу з нестисненою відеопослідовальністю;
- прискорення обчислювального процесу – відношення часу стиснення всіх тестових відеопослідовальностей на одному вузлі системи до часу стиснення на всіх задіяних вузлах системи;
- ефективність розподіленої системи – відношення прискорення обчислювального процесу до кількості задіяних вузлів.

Для організації експериментів використано шість загальноприйнятих для тестування методів стиснення відеопослідовальностей: три з них у форматі CIF (352x288 пікселів) [9], а три у форматі Ultra HD (3840x2160) [10] – «Foreman», «News», «Coastguard». Всі відеопослідовальності містять по 300 кадрів.

В розподіленій комп'ютерній системі на вузле-планувальнику встановлюється аналізатор відеопослідовальностей, приймаючий вхідну відеоінформацію і виконуючий її розподілення по вузлах комп'ютерної системи методом [7], при цьому чутливість аналізатора задається дуже високою. По мірі появи частин відеопослідовальностей стаються в чергу FIFO, направлену на обчислювальні вузли. Як тільки звільнюється вузол системи, він видає відеопослідовальність з черги, стискає її алгоритмами стандарту H.264 за допомогою вільно розповсюджуваного відкритого відеокодека FFmpeg [11] і повертає результат планує, де виконується збір і конкатенація стиснутих відеопослідовальностей. Якщо всі ресурси зайняті, ча-

сти відеопослідовальностей очікують в черзі, поки не звільниться хоча б один вузол. При цьому бар'єрна синхронізація не виконується, тобто процеси стиснення відеопослідовальностей вузлами відбуваються асинхронно і незалежно один від одного. Це дозволяє зменшити накладні витрати на очікування звільнення всіх замовлених в розподіленій системі ресурсів і синхронізацію вузлів системи.

В експериментах задіяно 4 логічних вузлів, наданих для досліджень кластера Інститута проблем моделювання в енергетиці (ІПМЭ) ім. Г.Е. Пухова г. Київ. Вузли побудовані на двох центральних процесорах Intel Xeon 5405 кожен з 8 Гб DDR-2. Комунікаційна середовище InfiniBand 20 Гб/с. Логічні вузли формуються з складу фізических вузлів. На вузлах встановлено планировщик Torque на ОС Scientific Linux 6.3.

Оцінка руху блоків для експериментів виконується двома алгоритмами: повний перебір, виконуючий пошук по всій області Ω , і один з алгоритмів «швидкий» оцінки руху, а саме, алгоритм «шестикутника», значно скорочуючий область пошуку. В результаті експериментів реєструється час стиснення кожної відеопослідовальності, загальний час обчислювального процесу, рівень искаження, коефіцієнт стиснення. Далі обчислюється прискорення обчислювального процесу і ефективність розподіленої системи.

3 РЕЗУЛЬТАТИ ІСЛІДОВАНИЙ

Для відеопослідовальностей у форматі CIF встановлено бітрейт 1 Мбіт/с, а для Ultra HD – 20 Мбіт/с. Для алгоритму повного перебору область пошуку встановлена 32 пікселі (на графіках позначається «ПП 32»), далі – 16 пікселів (на графіках – «ПП 16»). Для алгоритму «шестикутника» область пошуку – 32 пікселі (на графіках позначається «АШ 32»), далі – 16 пікселів («АШ 16»). Всі інші параметри кодера встановлені за замовчуванням для стандарту H.264. Залежність часу стиснення відеопослідовальностей від алгоритму з вибраною шириною області пошуку показана на рис. 1.

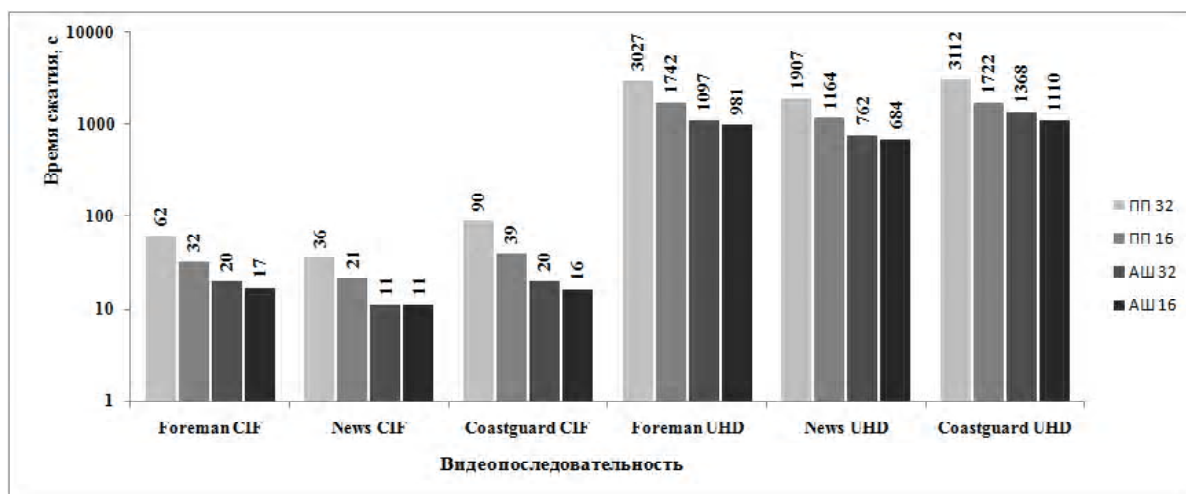


Рис. 1. Залежність часу стиснення відеопослідовальностей від алгоритму ОД з вибраною шириною області пошуку

В таблиці 1 показані характеристики ефективності розподіленої системи, використаної для сжатия видеопоследовательностей: T – общее время вычислительного процесса; T_n – накладные затраты на пересылки; T_a – накладные затраты на анализ и распределение частей видеопоследовательности; S – ускорение вычислительного процесса; E – эффективность распределенной системы.

Зависимость уровня искажения сжатых видеопоследовательностей относительно исходных, от алгоритма с выбранной шириной области поиска показана на рис. 2. На рис. 3 видно как влияет алгоритм ОД видеопоследовательности на коэффициент сжатия.

Таблиця 1. Характеристики ефективності використаної розподіленої системи

| Алгоритм ОД | T , с | T_n , % | T_a , % | S | E |
|-------------|---------|-----------|-----------|------|------|
| ПП 32 | 4003 | 15,71 | 7,11 | 2,21 | 0,53 |
| ПП 16 | 2651 | 23,73 | 10,57 | 2,02 | 0,51 |
| АШ 32 | 2277 | 27,62 | 12,3 | 1,72 | 0,43 |
| АШ 16 | 2019 | 31,15 | 13,87 | 1,71 | 0,43 |

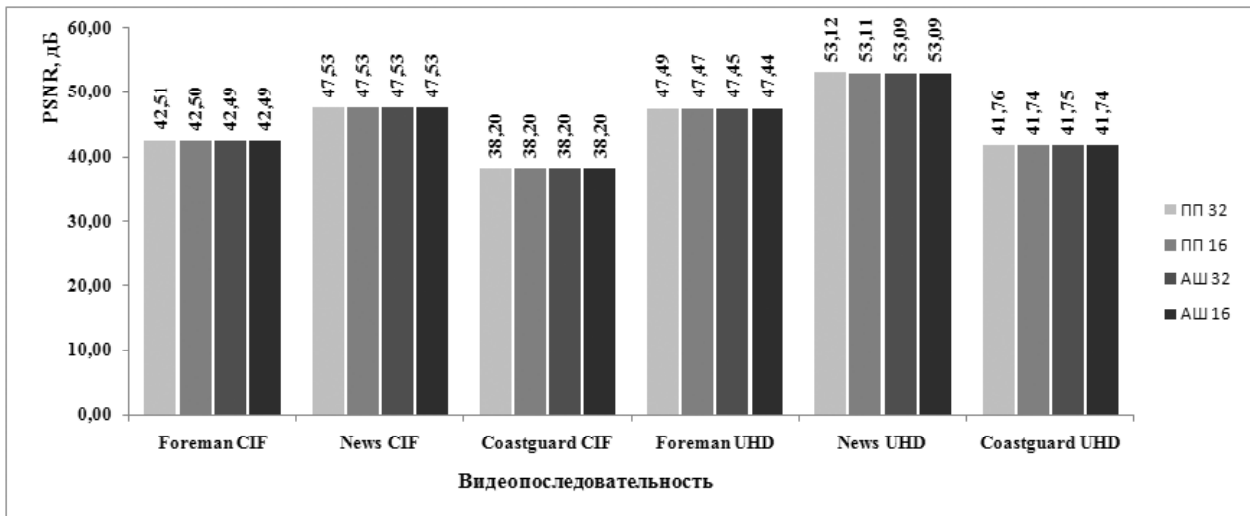


Рис. 2. Зависимость уровня искажения сжатых видеопоследовательностей от алгоритма ОД с выбранной шириной области поиска

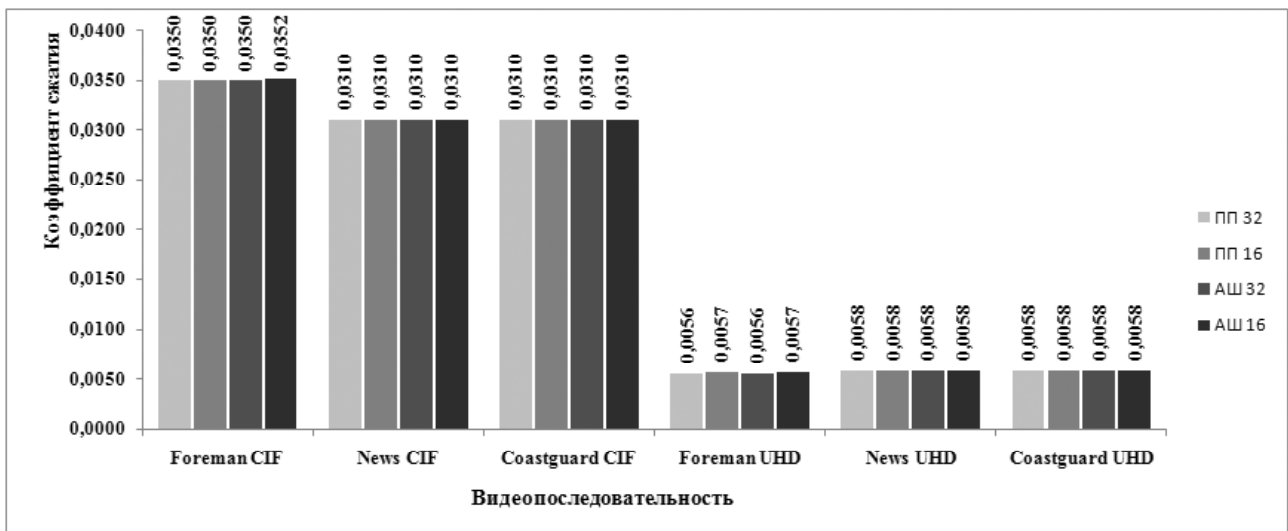


Рис. 3. Зависимость коэффициента сжатия видеопоследовательностей от алгоритма ОД с выбранной шириной области поиска

ВЫВОДЫ

В результате экспериментального исследования процесса сжатия Ultra HD-видеопоследовательностей в распределенной системе можно сформулировать следующие выводы:

- время сжатия видеопоследовательностей алгоритмом полного перебора векторов движения линейно зависит от разрешения кадра и ширины области поиска;

- в алгоритмах «быстрой» оценки движения, в частности, в алгоритме «шестиугольника» сокращение области поиска лишь незначительно уменьшает время сжатия;

- время сжатия видеопоследовательностей без интенсивного движения (например, «News») значительно короче времени сжатия видеопоследовательностей, содержащих интенсивное движение объектов («Foreman», «Coastguard»). Это связано с применением оптимизирующих алгоритмов в стандарте H.264, позволяющих пропускать поиск векторов движения для смежных кадров, не содержащих движение блоков;

- уровень искажения соответствующих видеопоследовательностей, предварительно разделенных на части методом [7] и распределенных по узлам компьютерной системы, изменяется незначительно в зависимости от ширины области поиска в алгоритме ОД. Это обусловлено тем, что после разделения на части смежные кадры таких видеопоследовательностей характеризуются значительной корреляционной зависимостью, следовательно, вектора движения не выходят за ближайшие блоки и оценка движения может быть выполнена на небольшой области поиска;

- алгоритм «быстрой» оценки движения демонстрирует незначительное снижение качества результата (по метрике PSNR) при существенном сокращении времени сжатия по сравнению с алгоритмом полного перебора;

- коэффициент сжатия соответствующих видеопоследовательностей практически не зависит от ширины области поиска векторов движения и от применяемого алгоритма ОД;

- время, затраченное на транспортировку видеопоследовательностей в формате Ultra HD в распределенной системе, составило до 31,15 % от всего вычислительного процесса и существенно снизило ускорение процесса сжатия и эффективность системы. Это является значительным фактором, влияющим на производительность вычислительного процесса;

- накладные затраты на анализ видеопоследовательностей методом [7] прямо пропорциональны разрешению кадра видеопоследовательности и количеству таких кадров. Для видеопоследовательностей в формате Ultra HD накладные затраты на анализ приемлемы и составляют до 13,87 % времени всего вычислительного процесса;

- при сжатии видеопоследовательностей в формате Ultra HD алгоритмы стандарта H.264 продемонстрировали приемлемые показатели коэффициента сжатия и уровня искажения, однако, время сжатия таких видеопоследовательностей слишком велико. В новом стандарте H.265, принятом сектором стандартизации электро-

связи Международного союза электросвязи ИТУ-Т в апреле 2013 года предусматривается повышение эффективности сжатия видеoinформации (снижение коэффициента сжатия при том же уровне искажения, который демонстрируют алгоритмы стандарта H.264) в том числе за счет усложнения алгоритмов оценки движения видеопоследовательностей [12].

Представляется перспективным исследование, направленное на снижение вычислительной сложности алгоритмов оценки движения видеопоследовательностей, применительно к видеoinформации в формате Ultra HD.

СПИСОК ЛИТЕРАТУРЫ

1. Ultra HD TVs stole the show at CES 2013, but they're just part of the puzzle / Engadget news. – Режим доступа: \www/ URL: <http://www.engadget.com/2013/01/11/ultra-hd-tvs-stole-the-show-at-ces-2013>. – Загл. с экрана.
2. Скрупский, С. Ю. Методы компрессии видеoinформации / С. Ю. Скрупский // Наукові праці Донецького національного технічного університету. Серія: «Обчислювальна техніка та автоматизація». – 2011. – № 21 (183). – С. 122–130.
3. Generic Framework for Parallel and Distributed Processing of Video-Data : in 4th International Symposium on Parallel and Distributed Processing and Applications (ISPA), vol. LNCS 4331 / D. Farin, P. With. – Sorrento, Italy, 2006. – P. 823–832.
4. A Parallel implementation of H.26L video encoder : in proc. of EuroPar 2002 conf. (LNCS 2400) / J. C. Fern6ndez, M. P. Malumbres – Padderborn, 2002. – P. 830–833.
5. Performance Evaluation of Parallel MPEG-4 Video Coding Algorithms on Clusters of Workstations : PARELEC '04 Proceedings of the international conference on Parallel Computing in Electrical Engineering / A. Rodriguez, A. Gonzalez, M. P. Malumbres. – Washington DC, 2004. – P. 354–357.
6. Скрупский, С. Ю. Повышение производительности распределенных систем компрессии видеoinформации / С. Ю. Скрупский // Науковий вісник Чернівецького національного університету імені Юрія Федьковича. Серія: Комп'ютерні системи та компоненти. – 2011. – №4, Т. 2. – С. 69–79.
7. Скрупский, С. Ю. Повышение эффективности сжатия видеoinформации в распределенных компьютерных системах / С. Ю. Скрупский // Электронное моделирование. – 2011, №6 (33). – С. 57–72.
8. Ватолин, Д. Методы для объективной оценки качества видеокодексов по сжатым ими видеопоследовательностям : материалы девятого научно-практического семинара «Новые информационные технологии в автоматизированных системах», март 2006 г. Москва / Д. Ватолин, А. Паршин. – М., 2006. – С. 4–12.
9. 4K Test Sequences. Reviving the classics. – Режим доступа: \www/ URL: <http://www.elementaltechnologies.com/resources/4k-test-sequences>. – Загл. с экрана.
10. YUV Video Sequences. – Режим доступа: \www/ URL: <http://trace.eas.asu.edu/yuv/>. – Загл. с экрана.
11. FFmpeg is a complete, cross-platform solution to record, convert and stream audio and video. – Режим доступа: \www/ URL: <http://www.ffmpeg.org/>. – Загл. с экрана.
12. ITU-T H.265 High efficiency video coding. – Режим доступа: \www/ URL: <http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=11885>. – Загл. с экрана.

Стаття надійшла до редакції 17.09.2013.

Скрупський С. Ю.

Канд. техн. наук, старший викладач, Запорізький національний технічний університет, Україна

КОМПРЕСІЯ ULTRA HD-ВІДЕОІНФОРМАЦІЇ У РОЗПОДІЛЕНІЙ КОМП'ЮТЕРНІЙ СИСТЕМІ

Розглянуто переваги та недоліки сучасного формату відтворення відеоінформації Ultra HD. Виконано оцінку обчислювальної складності операції пошуку векторів руху у відеопослідовності. Проаналізовано метод розподілу відеоінформації по вузлах комп'ютерної системи, який використано для розпаралелювання процесу компресії відеоінформації на кластері. Експериментально досліджено об'єктивні показники результату компресії Ultra HD-відеоінформації у розподіленій системі в залежності від використовуваного алгоритму оцінки руху та ширини області пошуку векторів руху.

Ключові слова: Ultra HD-відеоінформація, розподілена комп'ютерна система, компресія, бітрейт, коефіцієнт ущільнення, рівень викривлення, кореляція.

Skrupsky S. Y.

Ph.D., senior lecturer, Zaporizhian National Technical University, Ukraine

ULTRA HD-VIDEOINFORMATION COMPRESSION IN DISTRIBUTED COMPUTER SYSTEM

The article deals with the parallelization of Ultra HD-videoinformation compression process in distributed system. The advantages and disadvantages of modern Ultra HD video format have been discussed. The evaluation of the computational complexity of the motion estimation operation in video sequences has been performed. The method for video distribution over the nodes of a distributed system, based on correlation analysis of video frames with the dynamic sensitivity function, has been analyzed. It has been used for parallelization of videoinformation compression process at Pukhov Institute for Modelling in Energy Engineering cluster. The objective indicators of Ultra HD-videoinformation compression results in a distributed system depending on used motion estimation algorithm and the width of the motion vectors search region have been investigated experimentally. The overheads of videoinformation compression process in distributed system have been analyzed.

Keywords: Ultra HD-videoinformation, distributed computer system, compression, bit rate, compression ratio, distortion level, correlation.

REFERENCES

1. Ultra HD TVs stole the show at CES 2013, but they're just part of the puzzle / Engadget news. – Access mode: \www/ URL: <http://www.engadget.com/2013/01/11/ultra-hd-tvs-stole-the-show-at-ces-2013>. – Title from screen.
2. Skrupskij S. Yu. Metody kompressii videoinformacii. *Naukovi pratsi Donetskoho natsionalnoho tekhnichnoho universytetu. Serii: «Obchysliuvalna tekhnika ta avtomatyzatsiia»*, 2011, No. 21 (183), pp. 122–130.
3. Farin D., With P. Generic Framework for Parallel and Distributed Processing of Video-Data : in 4th International Symposium on Parallel and Distributed Processing and Applications (ISPA), vol. LNCS 4331, Sorrento, Italy, 2006, pp. 823–832.
4. Fernández J. C., Malumbres M. P. A Parallel implementation of H.26L video encoder : in proc. of EuroPar 2002 conf. (LNCS 2400), Paderborn, 2002. pp. 830–833.
5. Rodriguez A., Gonzalez A., Malumbres M. P. Performance Evaluation of Parallel MPEG-4 Video Coding Algorithms on Clusters of Workstations : PARELEC '04 Proceedings of the international conference on Parallel Computing in Electrical Engineering. Washington DC, 2004, pp. 354–357.
6. Skrupskij S. Yu. Povyshenie proizvoditel'nosti raspredeleennyx sistem kompressii videoinformacii. *Naukovyi visnyk Chernivetskoho natsionalnoho universytetu imeni Yurii Fedkovycha. Serii: Kompiuterni systemy ta komponenty*, 2011, No. 4, Vol. 2, pp. 69–79.
7. Skrupskij S. Yu. Povyshenie e'fektivnosti szhatiya videoinformacii v raspredeleennyx komp'yuternyx sistemax. *E'lektronnoe modelirovanie*, 2011, No. 6 (33), pp. 57–72.
8. Vatolin D., Parshin A. Metody dlya ob'ektivnoj ocenki kachestva videokodekov po szhatym imi videoposledovatel'nostyam : materialy devyatogo nauchno-prakticheskogo seminaru «Novye informacionnye texnologii v avtomatizirovannyx sistemax», mart 2006, Moskva, 2006, pp. 4–12.
9. 4K Test Sequences. Reviving the classics. – Access mode: \www/ URL: <http://www.elementaltechnologies.com/resources/4k-test-sequences>. – Title from screen.
10. YUV Video Sequences. – Access mode: \www/ URL: <http://trace.eas.asu.edu/yuv/>. – Title from screen.
11. FFmpeg is a complete, cross-platform solution to record, convert and stream audio and video. – Access mode: \www/ URL: <http://www.ffmpeg.org/>. – Title from screen.
12. ITU-T H.265 High efficiency video coding. – Access mode: \www/ URL: <http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=11885>. – Title from screen.