

ГИБРИДНЫЙ АЛГОРИТМ ДЛЯ БЫСТРОЙ ОЦЕНКИ ДВИЖЕНИЯ В СТАНДАРТЕ H.265/HEVC

ДОАН ТИЕН БАН, А. А. ТРОПЧЕНКО

*Университет ИТМО, 197101, Санкт-Петербург, Россия
E-mail: zayka_98rus@mail.ru*

Рассмотрены быстрые алгоритмы, предназначенные для сокращения вычислительной сложности межкадрового кодирования при высокоэффективном кодировании видеоизображений в стандарте H.265. Предложенный метод оценки движения использует алгоритм поиска по тестовой зоне, раннее определение кодированного блока, режим быстрого кодирования и флаг кодированного блока в качестве индикатора для определения преимуществ прогнозирования. По сравнению с эталонным H.264/AVC, быстрые алгоритмы сокращают время кодирования в среднем на 47,07 %, скорость передачи данных на 1,27 %, при этом потеря качества составляет 0,11 дБ.

Ключевые слова: HEVC, поиск по тестовой зоне, оценка движения, межкадровое кодирование, быстрый метод

Введение. H.265 или HEVC (High Efficiency Video Coding) — формат сжатия видеоданных, разработанный в связи с растущей потребностью в более высокой степени сжатия движущихся изображений для самых разных приложений, таких как потоковая передача в сети Интернет, передача данных, видеоконференции, цифровые запоминающие устройства и телевизионное вещание сверхвысокой четкости [1]. По сравнению с H.264/AVC (Advanced Video Coding), в формате HEVC предусматривается сжатие данных, на 25—50 % превышающее первый при том же уровне качества видеоданных или существенно улучшенное их качество при той же скорости передачи данных [2]. Как и в H.264/AVC, оценка движения ME (Motion Estimation) и компенсация движения MC (Motion Compensation) являются наиболее сложной вычислительной процедурой. Оценка движения заключается в определении наилучшего согласованного блока в опорных кадрах (предыдущих или будущих кадрах) и создании вектора движения MV (Motion Vector), тогда как компенсация движения создает невязки между двумя блоками с использованием вектора MV. Из-за переменного размера блока, нескольких режимов прогнозирования, нескольких опорных кадров и их более высокого разрешения процедура ME занимает большую часть общего объема вычислений при кодировании в кодерах H.265/HEVC — более 80 %, как доказано в работе [3].

По мере увеличения сложности кодирования время кодирования также увеличивается. Для того чтобы его сократить, были предложены различные алгоритмы оценки движения [4—10], и несмотря на то, что существует несколько эффективных алгоритмов, алгоритм поиска по тестовой зоне TZS (Test Zone Search) остается конкурентоспособным.

В настоящей статье алгоритм TZS используется в качестве метода оценки движения. В целях дальнейшего сокращения времени кодирования авторами предложено несколько алгоритмов для оценки движения, параметры которых рассматриваются в статье.

Алгоритм TZS. TZS включает четыре основных этапа: выбор первоначального центра поиска (или прогнозирование вектора движения PMV — Prediction Motion Vector), поиск с использованием ромбовидного DS (Diamond Search) или квадратного шаблона SS (Square Search), растровый поиск RS (Raster Search) и растровое/звездовидное рафинирование. Эти этапы сводятся к следующему [11, 12].

Этап 1. Выбор первоначального центра поиска. Алгоритм TZS использует предсказатель вектора движения для прогнозирования начального центра поиска любого блока предсказания PU (Prediction Unit). Этот вектор выбирается на основе конкуренции между несколькими предикторами, в частности медианным, левым, верхним и верхним правым. Критерием сравнения служит сумма абсолютных разностей SAD (Sum of Absolute Differences) пикселей двух блоков. Предиктор, приводящий к самому низкому значению SAD, используется как центр поиска следующего этапа.

Этап 2. Ромбический или квадратичный поиск. Выполняется поиск с использованием ромбовидного или квадратного шаблона размерностью 8, как показано на рис. 1, а и б соответственно, с расстояниями от 1 до 64 (если длина поиска равна 64).

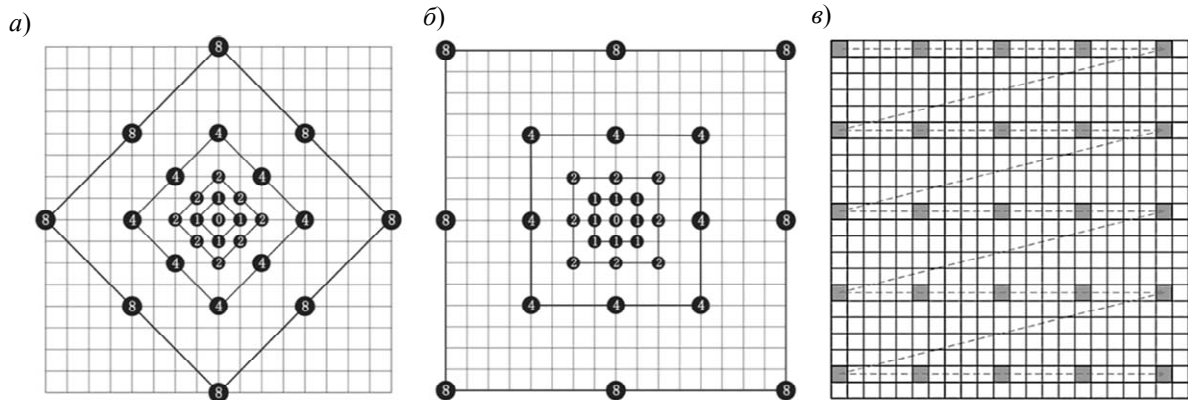


Рис. 1

Этап 3. Растровый поиск. Если расстояние между оптимальной точкой, определенной на этапе 2, и текущим центром поиска, которое называется наилучшим расстоянием L , является оптимальными (равно нулю), то поиск останавливается. Если $1 < L < R$ (где R — заданное пороговое значение количества пикселей), то осуществляется переход к рафинированию (этап 4). В противном случае ($L > R$) растровое сканирование выполняется с использованием значения R в качестве длины шага. Этот растровый поиск представляет собой полный поиск FS (Full Search) при $R = 5$ (рис. 1, в).

Этап 4. Растровое/звездовидное рафинирование. Оптимальная точка, определенная на этапе 3, выбирается в качестве начальной. Рафинирование может быть выполнено с использованием ромбовидного или квадратного шаблона с разными расстояниями — от 1 до диапазона поиска — путем умножения расстояния L на два в каждой итерации (звездовидное) или деления L на два в каждой итерации до достижения $L=1$. При $L=1$ осуществляется поиск двух соседних точек для выбора наилучшего оцененного вектора движения, имеющего наилучшую сумму абсолютной разности.

Быстрая конфигурация. Несмотря на удовлетворительный результат поиска, временные характеристики алгоритма TZS все еще низки. Для сокращения времени кодирования следует использовать комбинацию быстрых решений, таких как раннее завершение единицы кодирования ECU (Early CU termination), раннее обнаружение пропуска ESD (Early Skip Detection) и быстрый алгоритм решения флага кодированного блока CFM (Coded Block Flag Fast Mode Decision).

Алгоритм ECU позволяет использовать раннее определение блока кодирования CU (Coding Unit), пропущенные CU не будут разделяться далее.

При определении наилучшего размера CU формат HEVC проверяет каждый возможный размер CU, чтобы оценить эффективность кодирования. Это вызывает большую вычислительную сложность в процессе кодирования, которая должна быть преодолена для реализации быстрого кодера. В кодере НМ ключевой функцией для обработки CU является

TEncCU::xCompressCU, которая оценивается в различных режимах прогнозирования. Быстрое определение глубины CU реализуется с учетом того, что дополнительная обработка поддеревьев кодирования не требуется, когда текущий узел выбирает режим прогнозирования SKIP в качестве лучшего режима (см. рис. 2).

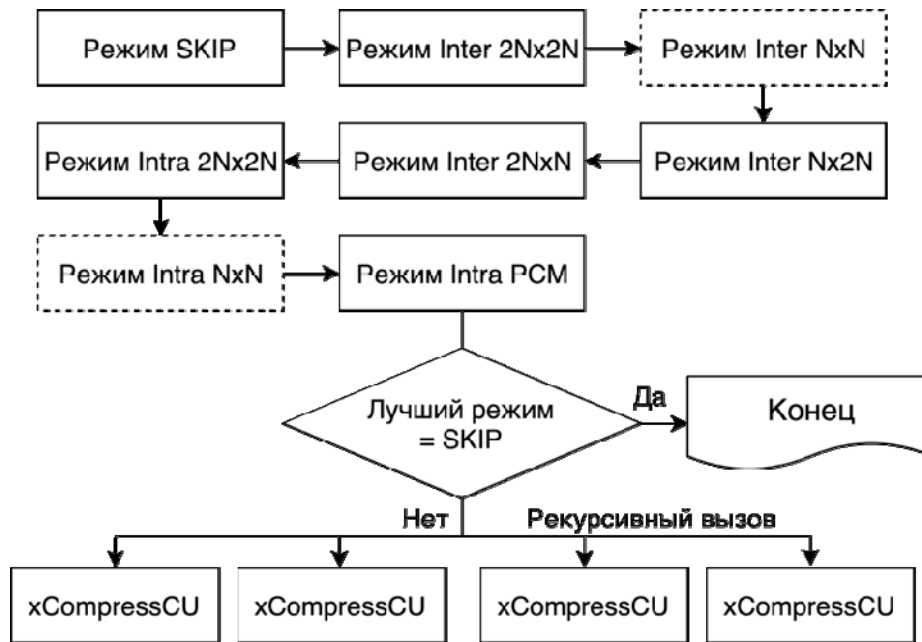


Рис. 2

Метод быстрого решения алгоритма ECU в коде HEVC путем раннего определения размера CU на основе сокращения дерева кодирования позволяет снизить время кодирования приблизительно на 42 % по сравнению с кодером HM-3.1 с незначительной потерей битрейта (менее 0,6 %) [13].

Алгоритм **ESD** позволяет использовать режим быстрого кодирования на основе флага кодированного блока.

Чтобы определить наилучший режим PU (быстрого прогнозирования), вычисляется значение скорости искажения RD (Rate-Distortion) во всех возможных режимах PU. Сложность кодирования может быть уменьшена путем проверки дифференциального вектора движения DMV (Differential Motion Vector) и флага кодированного блока CBF (Coded Block Flag) в результате поиска наилучшего режима Inter 2N×2N. Текущий алгоритм осуществляет поиск режимов Inter 2N×2N перед проверкой режима SKIP. После выбора наилучшего режима Inter 2N×2N с минимальным значением RD проверяются его DMV и CBF: если они соответственно равны (0, 0) и нулю (эти два условия называются „ранними условиями SKIP“), лучший режим текущего алгоритма определяется раньше, чем режим SKIP. Обнаружение режима SKIP на самой ранней возможной стадии позволяет сократить время кодирования в среднем на 33 % с очень незначительной потерей производительности кодирования (0,45 % по сравнению с HM-4.0) [14].

Возможна комбинация предложенных алгоритмов ECU и ESD, как показано на рис. 3.

Алгоритм **CFM** позволяет снизить сложность кодирования путем использования CBF в качестве индикатора для определения преимуществ прогнозирования. После выполнения преобразования и квантования в остаточном блоке, если его коэффициенты равны нулю, флаг устанавливается в нуль, в противном случае — в единицу. Флаг кодированного блока, равный нулю, указывает на достаточность прогнозирования, и кодирование остаточных коэффициентов не требуется [15].

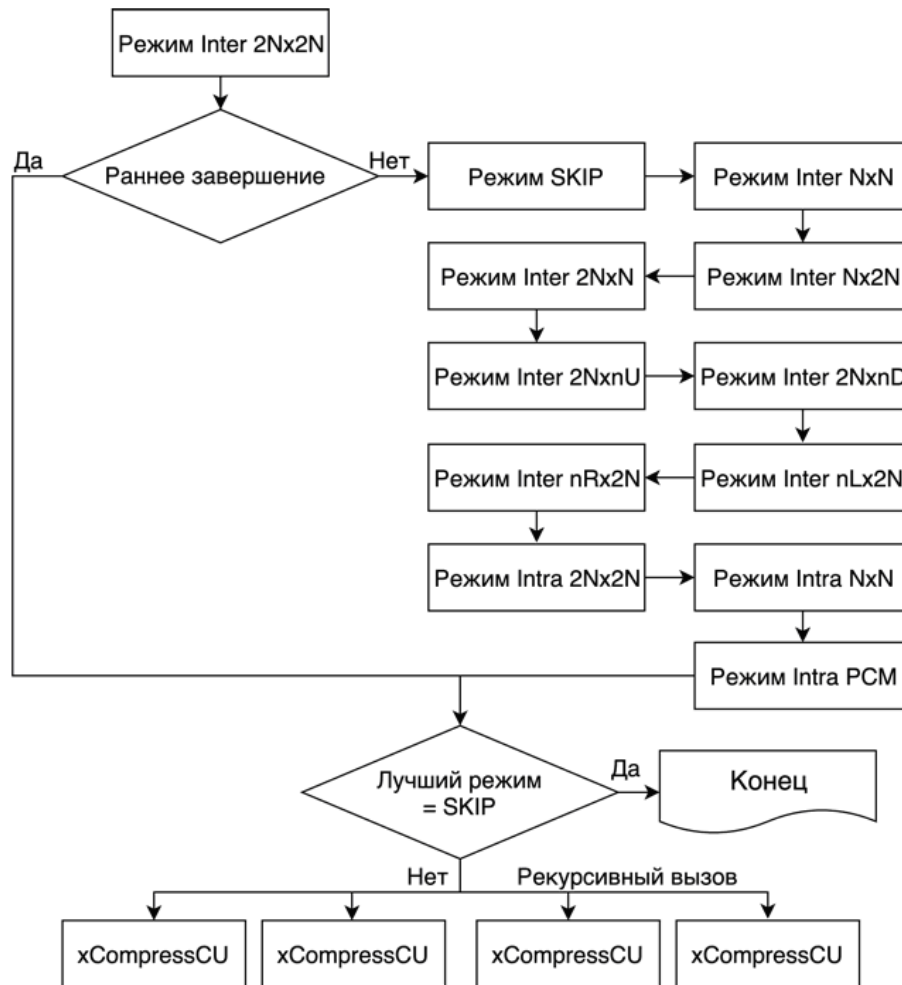


Рис. 3

Экспериментальные результаты. Предложенные быстрые алгоритмы были реализованы в эталонном программном обеспечении HEVC (HM-16.20) с конфигурацией основного профиля произвольного доступа RA (Random Access). Эксперимент проводился на платформе macOS 10.14.5 с процессором Intel (R) Core i5-5257U, тактовой частотой 2,7 ГГц и объемом оперативной памяти 8 ГБ. Для тестирования разработанных алгоритмов использован ряд стандартных видеопоследовательностей с четырьмя значениями параметра квантования QP (Quantization Parameter): 22, 27, 32, 37. Максимальный диапазон поиска составляет 64 с максимальным размером CU, равным 64, и максимальной глубиной CU, установленной на 4.

Для оценки эффективности предложенного метода комбинации быстрых алгоритмов был произведен их сравнительный анализ с эталонным программным обеспечением HM-16.20 по качеству сжатия PSNR (Peak Signal-to-Noise Ratio), скорости передачи и времени кодирования:

$$\Delta\text{PSNR} = \text{PSNR}_{\text{HM-16.20}} - \text{PSNR}_{\text{пр}};$$

$$\Delta\text{BR} = \frac{\text{BR}_{\text{HM-16.20}} - \text{BR}_{\text{пр}}}{\text{BR}_{\text{HM-16.20}}};$$

$$\Delta T = \frac{T_{\text{HM-16.20}} - T_{\text{пр}}}{T_{\text{HM-16.20}}},$$

где ΔPSNR и ΔBR — уменьшение показателя PSNR и скорости передачи данных (битрейта BR); ΔT — сокращение времени кодирования; индекс „пр“ соответствует предложенному быстрому методу.

Результаты сравнения оригинального и предложенного методов по показателю PSNR и скорости передачи данных представлены в табл. 1. Как видно, значение ΔBR уменьшено на 1,27 %, при этом потеря качества составила 0,11 дБ при использовании предлагаемого метода по сравнению с оригинальным HM-16.20.

Таблица 1

Разрешение	Видеопоследовательность	$\Delta PSNR$, дБ	ΔBR , %
2560×1600	Traffic	0,134	2,02
	PeopleOnStreet	0,121	0,92
1920×1080	Kimono	0,048	-0,03
	ParkScene	0,113	1,82
	Cactus	0,082	1,06
	BasketballDrive	0,036	0,74
	BQTerrace	0,074	1,81
832×480	BasketballDrill	0,084	1,21
	BQMall	0,140	1,39
	PartyScene	0,118	1,37
	RaceHorses	0,094	0,99
416×240	BasketballPass	0,171	1,62
	BQSquare	0,189	1,84
	BlowingBubbles	0,116	1,16
	RaceHorses	0,137	1,22
Среднее		0,110	1,27

Пример RD-зависимости для видеопоследовательности PeopleOnStreet исходного программного обеспечения HM-16.20 по сравнению с предложенным методом показан на рис. 4.

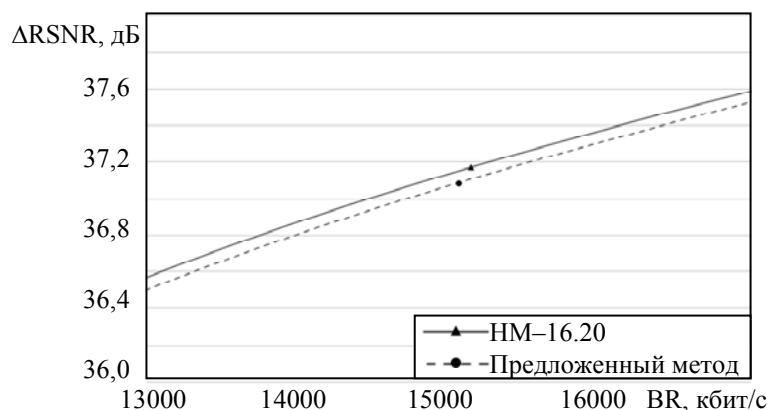


Рис. 4

Несмотря на то, что предложенный метод показывает небольшое повышение скорости передачи данных и даже незначительную потерю качества, его важной особенностью является сокращение времени кодирования. По сравнению с оригинальным программным обеспечением HM-16.20 предложенные быстрые алгоритмы выигрывают в среднем 47,07 % во времени кодирования (табл. 2).

Таблица 2

Разрешение	Видеопоследовательность	ΔT , %			
		QP = 22	QP = 27	QP = 32	QP = 37
2560×1600	Traffic	47,85	65,35	73,95	80,91
	PeopleOnStreet	15,09	27,51	32,79	46,45
1920×1080	Kimono	20,34	34,54	48,41	61,45
	ParkScene	41,28	58,04	69,28	77,97
	Cactus	36,38	48,46	57,06	66,26
	BasketballDrive	30,38	43,43	50,66	61,27
	BQTerrace	24,00	65,30	80,67	87,97

Продолжение табл. 2

Разрешение	Видеопоследовательность	$\Delta T, \%$			
		QP = 22	QP = 27	QP = 32	QP = 37
832×480	BasketballDrill	26,01	37,91	49,06	60,34
	BQMall	33,50	45,19	54,31	63,44
	PartyScene	21,44	34,39	45,02	57,58
	RaceHorses	11,95	25,35	34,96	45,52
1280×720	BasketballPass	45,42	55,62	65,01	72,43
	BQSquare	34,64	59,50	70,87	78,27
	BlowingBubbles	15,72	31,91	42,14	54,31
	RaceHorses	9,83	18,76	28,94	41,65
Среднее (по QP)		27,59	43,42	53,54	63,72
Среднее		47,07			

Заключение. На основе полученных результатов можно утверждать, что вычислительные затраты на оценку движения в стандарте H265/HEVC могут быть уменьшены путем использования ряда быстрых алгоритмов, таких как раннее завершение единицы кодирования, раннее обнаружение пропуска и быстрый алгоритм решения флага кодированного блока. Согласно экспериментам использование предложенного быстрого метода позволяет сократить время кодирования видеоданных в среднем на 47,07 % и снизить скорость передачи на 1,27 % при ухудшении качества сжатия всего на 0,11 дБ. Данный метод рекомендуется использовать в задачах сжатия видеопоследовательностей, где характеристики движения являются сложными и время кодирования критично.

СПИСОК ЛИТЕРАТУРЫ

1. Sullivan G. J., Ohm J.-R., Han W.-J., Wiegand T. Overview of the high efficiency video coding (HEVC) standard // IEEE Transact. Circuits Systems Video Technol. 2012. Vol. 22, N 12. P. 1649—1668. DOI: 10.1109/TCSVT.2012.2221191.
2. Ohm J.-R., Sullivan G. J., Schwarz H., Tan T. K., Wiegand T. Comparison of the coding efficiency of video coding standards—including high efficiency video coding (HEVC) // IEEE Transact. Circuits Systems Video Technol. 2012. Vol. 22, N 13. P. 1669—1684. DOI: 10.1109/TCSVT.2012.2221192.
3. Grellert M., Shafique M., Khan M. U. K. et. al. An adaptive workload management scheme for HEVC encoding // Proc. of IEEE Intern. Conf. on Image Processing. Australia. 2013. P. 1—5. DOI: 10.1109/ICIP.2013.6738381.
4. Jain J., Jain A. Displacement measurement and its application in interframe image coding // IEEE Transact. on Communications. 1981. Vol. 29, N 12. P. 1799—1808. DOI: 10.1109/TCOM.1981.1094950.
5. Zhu C., Lin X., Chau L.-P. Hexagon-based search pattern for fast block motion estimation // IEEE Transact. Circuits Systems Video Technol. 2002. Vol. 12, N 5. P. 349—355. DOI: 10.1109/TCSVT.2002.1003474.
6. Zhu S., Ma K.-K. A new diamond search algorithm for fast block matching motion estimation // Proc. of the Intern. Conf. on Information, Communications and Signal Processing. Singapore. 1997. P. 292—296. DOI: 10.1109/ICICS.1997.647106.
7. Yin P., Tourapis H.-Y. C., Tourapis A. M., Boyce J. Fast mode decision and motion estimation for JVT/H.264 // Proc. of the Intern. Conf. on Image Processing. Spain. 2003. Vol. 3. P. 853—856. DOI: 10.1109/ICIP.2003.1247379.
8. Xu X., He Y. Comments on motion estimation algorithms in current JM software (JVT-Q089) // Joint Video Team Document, 17th Meeting. Nice, France, 2005.
9. Нгуен Ван Чьонг, Тропченко А. А. Иерархический адаптивный алгоритм шаблонного поиска для оценки движения при анализе видеопоследовательности // Научно-технический вестник информационных технологий, механики и оптики. 2016. Т. 16, № 3. С. 474—481. DOI: 10.17586/2226-1494-2016-16-3-474-481.
10. Нгуен Ван Чьонг, Тропченко А. А. Быстрый алгоритм поиска по тестовой зоне для межкадрового кодирования // Научно-технический вестник информационных технологий, механики и оптики. 2017. Т. 17, № 3. С. 483—489. DOI: 10.17586/2226-1494-2017-17-3-483-489.

11. Belghith F., Kibeya H., Loukil H., Ben Ayed M. A., Masmoudi N. A new fast motion estimation algorithm using fast mode decision for high-efficiency video coding standard // *J. of Real-Time Image Processing*. 2016. Vol. 11, N 4. P. 675—691. DOI: 10.1007/s11554-014-0407-0.
12. Kibeya H., Belghith F., Loukil H., Ben Ayed M. A., Masmoudi N. TZSearch pattern search improvement for HEVC motion estimation modules // *Proc. of the Intern. Conf. on Advanced Technologies for Signal and Image Processing*. Tunisia. 2014. P. 95—99. DOI: 10.1109/ATSIP.2014.6834584.
13. Choi K., Park S.-H., Jang E. S. Coding tree pruning based CU early termination // Document JCTVC-F092, JCT-VC. Torino, Italy. July 2011.
14. Yang J., Kim J., Won K., Lee H., Jeon B. Early SKIP detection for HEVC // Document JCTVC-G543, JCT-VC. Geneva, Switzerland. Nov. 2011.
15. Bossen F., Flynn D., Sharman K., Suhring K. HM Software Manual. 2018.

Сведения об авторах

Доан Тиен Бан

— аспирант; Университет ИТМО; факультет программной инженерии и компьютерной техники; E-mail: bandoan@mail.ru

Андрей Александрович Тропченко

— канд. техн. наук, доцент; Университет ИТМО; факультет программной инженерии и компьютерной техники; E-mail: zayka_98rus@mail.ru

Поступила в редакцию
24.10.19 г.

Ссылка для цитирования: Доан Тиен Бан, Тропченко А. А. Гибридный алгоритм для быстрой оценки движения в стандарте H.265/HEVC // *Изв. вузов. Приборостроение*. 2020. Т. 63, № 1. С. 18—25.

HYBRID ALGORITHM FOR RAPID MOTION ESTIMATION IN THE H. 265/HEVC STANDARD

Doan Tien Ban, A. A. Tropchenko

ITMO University, 197101, St. Petersburg, Russia

E-mail: zayka_98rus@mail.ru

Fast algorithms aimed at reduction of computational complexity of inter-frame coding in the modern high efficiency video coding H.265 are considered. Algorithm Test Zone Search (TZS) is used as a method for motion estimation. The proposed method of motion estimation uses a search algorithm for the test zone, early coded block detection, fast encoding mode, and the coded block flag as an indicator to determine the advantages of forecasting. Compared to the reference HM-16.20, fast algorithms reduce the encoding time by an average of 47,07 %, the bitrate by 1,27 %, while the loss of quality is 0,11dB.

Keywords: HEVC, Test Zone Search, motion estimation, inter-frame coding, fast method

REFERENCES

1. Sullivan G.J., Ohm J.-R., Han W.-J., Wiegand T. *IEEE Trans. Circuits Syst. Video Technol.*, 2012, no. 12(22), pp. 1649–1668, DOI: 10.1109/TCSVT.2012.2221191.
2. Ohm J.-R., Sullivan G.J., Schwarz H., Tan T.K., Wiegand T. *IEEE Trans. Circuits Systems Video Technol.*, 2012, no. 13(22), pp. 1669–1684, DOI: 10.1109/TCSVT.2012.2221192.
3. Grellert M., Shafique M., Khan M.U.K. et al. *Proc. of IEEE International Conference on Image Processing*, Australia, 2013, pp. 1–5, DOI: 10.1109/ICIP.2013.6738381
4. Jain J., Jain A. *IEEE Transaction on Communications*, 1981, no. 12(29), pp. 1799–1808, DOI: 10.1109/TCOM.1981.1094950.
5. Zhu C., Lin X., Chau L.-P. *IEEE Trans. Circuits Syst. Video Technol.*, 2002, no. 5(12), pp. 349–355, DOI: 10.1109/TCSVT.2002.1003474.
6. Zhu S., Ma K.-K. *Proc. of International Conference on Information, Communications and Signal Processing*, Singapore, 1997, pp. 292–296, DOI: 10.1109/ICICS.1997.647106.
7. Yin P., Tourapis H.-Y.C., Tourapis A.M., Boyce J. *Proc. of International Conference on Image Processing*, Spain, 2003, vol. 3, pp. 853–856, DOI: 10.1109/ICIP.2003.1247379.
8. Xu X., He Y. *Joint Video Team Document*, 17th Meeting, Nice, France, 2005.
9. Truong Nguyen Van, Tropchenko A.A. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2016, no. 3(16), pp. 474–481, DOI: 10.17586/2226-1494-2016-16-3-474-481 (in Russ.)
10. Truong Nguyen Van, Tropchenko A.A. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2017, no. 3(17), pp. 483–489, DOI: 10.17586/2226-1494-2017-17-3-

- 483-489 (in Russ.)
11. Belghith F., Kibeya H., Loukil H., Ben Ayed M.A., Masmoudi N. *Journal of Real-Time Image Processing*, 2016, no. 4(11), pp. 675–691, DOI: 10.1007/s11554-014-0407-0.
 12. Kibeya H., Belghith F., H. Loukil H., Ben Ayed M.A., Masmoudi N. *Proc. of International Conference on Advanced Technologies for Signal and Image Processing*, Tunisia, 2014, pp. 95–99, DOI: 10.1109/ATSIP.2014.6834584.
 13. Choi K., Park S.-H., Jang E.S. *Document JCTVC-F092, JCT-VC*, Torino, Italy, Jul. 2011.
 14. Yang J., Kim J., Won K., Lee H., Jeon B. *Document JCTVC-G543, JCT-VC*, Geneva, Switzerland, Nov. 2011
 15. Bossen F., Flynn D., Sharman K., Suhring K. *HM Software Manual*, Sep. 2018.

Data on authors**Doan Tien Ban**

— Post-Graduate Student; ITMO University, Faculty of Software Engineering and Computer Systems; E-mail: bandoan@mail.ru

Andrey A. Tropchenko

— PhD, Associate Professor; ITMO University, Faculty of Software Engineering and Computer Systems; E-mail: zayka_98rus@mail.ru

For citation: Doan Tien Ban, Tropchenko A. A. Hybrid algorithm for rapid motion estimation in the H. 265/HEVC standard. *Journal of Instrument Engineering*. 2020. Vol. 63, N 1. P. 18—25 (in Russian).

DOI: 10.17586/0021-3454-2020-63-1-18-25