Національна академія наук України
Інститут кібернетики імені В.М. Глушкова

Кваліфікаційна наукова
праця на правах рукопису

Луц Василь Костянтинович

УДК 004.627, 004.932

ДИСЕРТАЦІЯ

МЕТОДИ І АЛГОРИТМИ ДЛЯ АДАПТИВНОГО КОДУВАННЯ ЗОБРАЖЕНЬ

113 – «Прикладна математика»
Галузь знань 11 – «Математика та статистика»
Подається на здобуття наукового ступеня доктора філософії.

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Луц В.К.

Науковий керівник Гнатів
Лев Олексійович,
к.т.н., с.н.с.

Київ – 2024

АНОТАЦІЯ

Луц В.К. Методи та алгоритми для адаптивного кодування зображень. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії за спеціальністю 113 «Прикладна математика». – Інститут кібернетики імені В.М. Глушкова Національної академії наук України, Київ. – 2024.

Об'єкт дослідження **–** підходи, методи і алгоритми кодування та стиснення зображень з втратами, комплекси адаптивних та фіксованих алгоритмів.

Мета роботи **–** дослідження та розробка підходів, методів, алгоритмів і адаптивних наборів алгоритмів кодування і стиснення зображень з втратами.

Мета роботи обумовлює дослідження наступних типів кодеків зображень: швидкодійні, збалансовані, ресурсоємні, універсальні. В свою чергу це визначає наступні завдання: дослідити адаптивні та фіксовані алгоритми для підвищення швидкодії; порівняти комплекси адаптивних алгоритмів; побудувати математичну модель кодування і стиснення зображень з втратами; розробити нові адаптивні режими та критерії їх вибору; розвинути нові підходи, методи та алгоритми адаптивного кодування і стиснення зображень з втратами.

Основні отримані результати: нова математична модель кодування і стиснення зображень з втратами; нова архітектура кодека, що включає додатковий структурний рівень – наборів алгоритмів, адаптивних режимів; концепція адаптивних кодеків різного рівня, оптимальних для певних обчислювальних обмежень; нові адаптивні режими, що мають значно менші вимоги до мінімально необхідних обчислювальних ресурсів кодування зображень і відео; цілочислове просте косинусне перетворення перетворення порядку 16 зі швидкими алгоритмами обчислення низької складності без виконання операцій множення; цілочислове однонормове модифіковане синус-косинусне перетворення типу VII порядку 8 зі швидкими алгоритмами обчислення; новий режим роздільного адаптивного кодування з вибором з трьох ортогональних перетворень (косинусного, синус-косинусного і синусного); два алгоритми передбачення значень пікселів на основі зовнішніх (для прогресивної

передачі) і внутрішніх даних (з метою зменшення обчислювальної складності); новий алгоритм вибору порядку і типу перетворень в залежності від коефіцієнта стиснення і статистичних характеристик вхідних даних; новий підхід, що об'єднує переваги фіксованого і адаптивного вибору алгоритмів. Для швидкодійного кодека, аналога H.263, що використовує фіксовані алгоритми, запропоновано наступні модифікації: використання одного універсального алгоритму передбачення значень пікселів, використання однонормових цілочислових перетворень порядку 8 та 16 зі швидкими алгоритмами обчислень зі спрощеним алгоритмом вибору одного перетворення для всього зображення, новий алгоритм вибору перетворення для всього зображення, алгоритм вибору перетворення в залежності від коефіцієнта стиснення, цілочислове просте косинусне перетворення порядку 16 низької складності без виконання операцій множення. В результаті отримаємо швидкодійний кодек, що потребує менше обчислювальних ресурсів ніж існуючий кодек H.263, але з коефіцієнтом стиснення як у H.264. Для кодека наступного рівня складності, збалансованого, запропоновано адаптивне використання перетворень порядку 8, 16 та 32, з попереднім вибором одного з наборів перетворень для низької, середньої та високої кореляції даних. В результаті отримаємо зменшення обчислювальної складності та збільшення коефіцієнта стиснення відносно існуючого H.264. Для ресурсоємного кодека запропоновано режим використання роздільних адаптивних перетворень трьох типів: косинусного, синусного і синус-косинусного, з адаптивним вибором режиму з двох чи трьох типів перетворень. Для всіх типів кодеків обгрунтовано доцільність використання прогресивної передачі даних і універсального алгоритму передбачення значень пікселів. Вперше запропоновано модифікація MSE, яка враховує лише значення, більші за обране порогове X. На основі модифікованої MSE обчислюється модифіковане PSNR, яке має підвищену точність відображення спотворень, особливо контурів та перепадів значень, а також значно зменшену обчислювальну складність. Вперше для простору кольорів розроблена нова векторна міра, що враховує як зміни яскравості, так і зміни кольорів. Запропонована міра може

використовуватись як для оцінки похибок кодування, так і для оцінки відстані між кольорами, зокрема при побудові локальних динамічних просторів кольорів. Запропоновано адаптивні алгоритми вибору простору кольорів: індексованої палітри, RGB та різних режимів YcrCb, а також новий простір кольорів на підставі співвідношення кольорів.

Практичне значення отриманих результатів полягає в обгрунтуванні системи адаптивних кодеків різного рівня, оптимальних для наявних обчислювальних ресурсів, і в запропонованих алгоритмах для кодеків різного класу (швидкодійних, збалансованих, ресурсоємних). Запропоновано нові адаптивні режими низької складності та зворотня сумісність кодеків.

Теоретичне значення отриманих результатів полягає в побудові математичної моделі кодування і стиснення зображень та відео з втратами, розробці нової концепції адаптивних кодеків різного рівня складності, нової архітектури кодеків з додатковим структурним рівнем вибору адаптивних режимів, розробці модифікованих мір похибок (MSE, PSNR та інш), нової векторної міри визначення дистанції між кольорами та похибки відновлення.

Результати досліджень були представлені в темах інституту кібернетики імені В.М. Глушкова НАН України, зокрема в наступних: ВФ 140.18, ВКМ 140.24.21, ВК.140.16.15, ВФ 140.14, ВП. 140.19, ВП. 140.21.

Було зроблено 11 публікацій, з них 6 статей, в тому числі 3 статті у періодичних наукових виданнях, проіндексованих у базах даних Scopus. Результати дисертації викладено в матеріалах 5 міжнародних наукових конференцій та симпозіумів.

Ключові слова: оптимізація, адаптивні алгоритми, адаптивні режими, адаптивне кодування, математичне моделювання, адаптивні набори алгоритмів, швидкодійний кодек, швидкі алгоритми, архітектура кодека, цілочисельні перетворення, швидкі перетворення, прості косинусні перетворення, синус-косинусні перетворення, кластерний аналіз, прогресивна передача даних, зворотня сумісність кодеків, універсальний алгоритм передбачення.

ANNOTATION

Luts V.K. Methods and Algorithms for Adaptive Image Encoding. – Qualifying scientific work as a manuscript.

Dissertation for the degree of Doctor of Philosophy in the specialty 113 "Applied Mathematics". – V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv. – 2024.

The **object of research** – approaches, methods, and algorithms for encoding and compressing images with losses, sets of adaptive and fixed algorithms.

The **purpose of the work** is the study and development of approaches, methods, algorithms, and adaptive sets of algorithms for encoding and compressing images with losses.

The aim of the work determines the research of the following types of image codecs: high-performance, balanced, resource-intensive, universal. This, in turn, defines the following tasks: investigate adaptive and fixed algorithms to improve performance; compare complexes of adaptive algorithms; build a mathematical model of image encoding and compression; develop new adaptive algorithms and modes; advance new approaches, methods, and algorithms for adaptive encoding and compression of images with losses.

Main results obtained: a new mathematical model for lossy image encoding and compression; a new codec architecture that includes an additional structural level—sets of adaptive modes; a concept of adaptive codecs of different levels, optimal for certain computational constraints; new adaptive modes with significantly reduced minimum computational resource requirements for encoding images and videos; a 16-point integer simple cosine transform with fast, low-complexity algorithms that do not require multiplication operations; an 8-point integer unit-norm modified sine-cosine transform type VII with fast computation algorithms; a new mode of separate adaptive encoding with a choice of three orthogonal transforms (cosine, sine-cosine, and sine); two algorithms for predicting pixel values based on external (for progressive transmission) and internal data (to reduce computational complexity); a new algorithm

for selecting the order and type of transforms depending on the compression ratio and statistical characteristics of the input data; a new approach that combines the benefits of fixed and adaptive algorithm selection; a new simple algorithm for assessing the statistical characteristics of input data to select fixed encoding parameters for an image/frame. For a high-performance codec analogous to H.263 that uses fixed algorithms, the following modifications are proposed: using a single universal algorithm for predicting pixel values, utilizing new integer transforms of orders 8 and 16 with fast computation algorithms and a simplified algorithm for selecting a single transform for the entire image, a new algorithm for selecting transforms for the entire image depending on the compression ratio, and a low-complexity 16-point integer simple cosine transform that does not perform multiplication operations. As a result, we obtain a high-performance codec that requires fewer computational resources than the existing H.263 codec, but with a compression ratio like H.264. For the next level of complexity codec, balanced, the adaptive use of transforms of orders 8, 16, and 32 is proposed, with preliminary selection of one set of transforms for low, medium, and high data correlation. This results in reduced computational complexity and increased compression ratio compared to existing H.264. For resource-intensive codecs, a mode using separate adaptive transforms of three types: cosine, sine, and sine-cosine, with adaptive selection of mode from two or three types of transforms, is proposed. For the first time, an MSE modification is proposed that only considers values above a selected threshold X. Based on the modified MSE, a modified PSNR is calculated, which has increased accuracy in displaying distortions, especially contours and value jumps, and significantly reduced computational complexity. For the color space, a new vector measure has been developed that takes into account both changes in brightness and color changes. The proposed measure can be used both for assessing encoding errors and for measuring the distance between colors, particularly in the construction of local dynamic color spaces. Proposed adaptive algorithms for selecting color space: indexed palette, RGB, and various YcrCb modes; a new color space based on color ratios.

The practical significance of the results obtained lies in the substantiation of a system of adaptive codecs of different levels, optimized for available computational resources,

and in the proposed algorithms for codecs of various classes (high-performance, balanced, resource-intensive). New low-complexity adaptive modes and backward compatibility of codecs have been proposed.

The theoretical significance of the results obtained lies in the construction of a mathematical model for lossy image and video encoding and compression, the development of a new concept of adaptive codecs of varying complexity, a new codec architecture with an additional structural level for selecting adaptive modes, the development of modified error measures (MSE, PSNR, etc.), and a new vector measure for determining the distance between colors and recovery errors.

The research results were presented in the following themes of the V.M. Glushkov Institute of Cybernetics of the National Academy of Sciences of Ukraine: VF 140.18, VKM 140.24.21, VP. 140.19, VP. 140.21.

There were 11 publications, including 6 articles, of which 3 are in periodical scientific publications indexed in Scopus databases. The results of the dissertation were presented at 5 international conferences.

Keywords: optimization, adaptive algorithms, adaptive modes, adaptive encoding, mathematical modeling, adaptive algorithm sets, high-performance codec, fast algorithms, codec architecture, integer transforms, fast transforms, simple cosine transforms, sine-cosine transforms, cluster analysis, progressive data transmission, codec backward compatibility, universal prediction algorithm.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті у періодичних наукових виданнях, проіндексованих у базах даних Web of Science Core Collection та/або Scopus

1. Hnativ L.O., Luts V.K. Algorithms for Fast Implementation of 4-Point Integer Sine Type VII Transforms without Multiplication and Separable Directional Adaptive Transforms for Intra Prediction in Image/Video Coding. Cybern Syst Anal. 56, 159–170 (2020). https://doi.org/10.1007/s10559-020-00231-y
2. Hnativ L.O., Luts V.K. Integer Modified Sine-Cosine Transforms Type VII. A construction Method and Separable Directional Adaptive Transforms for Intra Prediction with 8 × 8 Chroma Blocks in Image/Video Coding. Cybern Syst Anal. 57, 155–164 (2021). https://doi.org/10.1007/s10559-021-00339-9
3. Luts, V.K. New Approaches and Methods of Adaptive Image Encoding. Cybern Syst. Anal. 60, 331–340 (2024). https://doi.org/10.1007/s10559-024-00673-8

Статті у наукових виданнях, включених на дату опублікування до переліку наукових фахових видань України за спеціальністю 113

1. Luts Ya.V., Luts V.K. About the Development of a High-Speed Simplified Image Codec. Cybernetics and Computer Technologies. 2021. 1. P. 61–66. (in Ukrainian) https://doi.org/10.34229/2707-451X.21.1.6
2. Луц В., Луц Я. Про використання модифікованого дельта-алгоритму для обробки та кодування зображень. Фізико-математичне моделювання та інформаційні технології, (36), 148-152. (2023). https://doi.org/10.15407/10.15407/fmmit2023.36.148

Статті у наукових виданнях, включених на дату опублікування до переліку наукових фахових видань України за іншими спеціальностями

1. Луц В.К. Просте цілочисельне косинусне перетворення порядка 16 для кодування сигналів і зображень. *Штучний інтелект*. 2014. № 4. С. 31–43. URI: http://dspace.nbuv.gov.ua/handle/123456789/85309

Матеріали конференцій:

1. Луц В.К. Адаптивний ітераційний метод для зменшення розрядності зображень і відеоданих/ Проблемно-наукова міжгалузева конференція «Юриспруденція та проблеми інформаційного суспільства» (ЮПІС – 2018), 23-25 травня 2018 р., м. Надвірна. С. 63-68.
2. Луц В.К., Луц Я.В. «Адаптивні режими кодеків зображень», X Міжнародна науково-практична конференція Глушковські читання", м. Київ, 2 грудня 2021. С. 113-117
   URI: http://glushkov.kpi.ua/GC21
3. Луц В. Комплексні підходи та моделі для кодування та обробки зображень. Міжн. наук.-пр. конф. «Інформаційні технології та комп'ютерне моделювання», ІТКМ-2022, 15 – 16 грудня 2022 р., м. Івано-Франківськ, с. 60-62.
   https://itcm.comp-sc.if.ua/2023/ITCM-22_Zbirnyk_final.pdf
4. Луц В.К. Перспективи застосування алгоритмів кластерізації для зменшення розмірності і складності математичних моделей. X Міжнародна науково-практична конференція: "Глушковські читання", м. Київ, 2 грудня 2021 р., с. 109-113.
   URI: http://glushkov.kpi.ua/GC21
5. Луц В.К. Нові адаптивні кодеки, режими та алгоритми кодування зображень. Міжнародна науково-практична конференція «Інформаційні системи та технології: результати і перспективи», м. Київ, КНУ імені Тараса Шевченка, 6 березня 2024 р., с. 278-282.
   URI:https://www.ist.fit.knu.ua/news/abstracts-ist2024

ЗМІСТ

Вступ ……. 15
1 Огляд відомих підходів, методів і алгоритмів кодування зображень ……21
1.1 Огляд відомих форматів кодування зображень ……25
1.2 Дискретні ортогональні перетворення ……..26
1.3 Алгоритми передбачення значень пікселів …….. ……27
1.4 Адаптивні алгоритми кодування зображень ……28
1.5 Співвідношення похибки до коефіцієнта стиснення і критерії якості зображень ……...29
1.6 Роздільні адаптивні перетворення …….……31
1.7 Прогресивна передача даних ……...32
1.8 Перспективи застосування нейромережевих методів для стиснення і відновлення зображень/кадрів з втратами ……..33
1.8.1 Застосування нейромереж для розробки нових підходів, методів і алгоритмів стиснення і відновлення зображень/кадрів ……33
1.8.2 Застосування нейромереж для наближеного відновлення початкових даних ……..34
1.8.3 Застосування нейромереж для оптимізації адаптивного використання алгоритмів стиснення і фільтрації зображень……35
1.9 Висновки до розділу ……35
2 Нова архітектура адаптивного кодування і стиснення зображень …..……37
2.1 Основні поняття і закономірності ……..…….. ……..37
2.2 Математична модель кодування і стиснення зображень ……39
2.3 Швидкодійний кодек стиснення зображень/відео …….……44
2.3.1 Універсальний алгоритм передбачення значень пікселів……..45
2.3.2 Спрощені перетворення зі швидкими алгоритмами ……52
2.3.3 Алгоритм вибору типу перетворення ……53
2.3.4 Спрощений вибір перетворень порядку 8 та 16……..54
2.3.5 Спрощені простори кольорів та вибір оптимального простору кольорів ……57

2.3.6 Екстремальні режими стиснення зображень з втратами ……..59
2.4 Збалансований кодек стиснення зображень/відео ……..60
2.4.1 Адаптивний вибір оптимальних перетворень ……..60
2.4.2 Адаптивні режими середньої і високої складності ……...62
2.4.3 Алгоритми вибору адаптивного режиму і нова архітектура кодеків……63
2.4.4 Адаптивні режими низької складності ……64
2.5 Ресурсоємний кодек стиснення зображень/відео ……..……65
2.5.1 Локальні адаптивні алгоритми ……66
2.5.2 Адаптивні режими та вибір оптимального ……67
2.5.3 Локальний адаптивний вибір типу перетворення ……..68
2.6 Універсальний кодек стиснення зображень/відео ……....69
2.6.1 Алгоритми вибору адаптивних режимів ……70
2.6.2 Зворотня сумісність кодеків різного рівня ……70
2.7 Висновки до розділу ……...71
3 Цілочислові перетворення для адаптивного кодування ..……74
3.1 Цілочислове просте косинусне перетворення порядку 16 ……..74
3.1.1 Метод побудови цілочислового простого косинусного перетворення порядку 16 ……..74
3.1.2 Алгоритм швидкого обчислення прямого цілочислового ступінчастого перетворення порядку 8 ……79
3.1.3 Алгоритм швидкого обчислення оберненого цілочислового ступінчастого перетворення порядку 8 ……80
3.1.4 Обчислювальна складність цілочислового простого косинусного перетворення порядку 16 ……...82
3.1.5 Тестування і порівняльний аналіз цілочислового простого косинусного перетворення порядку 16 ……83
3.2 Цілочислове синусне перетворення типу VII порядку 4 ……87
3.3 Цілочислове однонормове модифіковане синус-косинусне перетворення типу VII порядку 8 ……..……..94

3.3.1 Метод побудови цілочислового модифікованого синус-косинусного перетворення типу VII порядку 8……95
3.3.2 Алгоритм швидкого обчислення 8-точкового прямого модифікованого синус-косинусного перетворення типу VII ….…98
3.3.3 Алгоритм швидкого обчислення 8-точкового оберненого модифікованого синус-косинусного перетворення типу VII ……100
3.3.4. Обчислювальна складність цілочислового модифікованого синус-косинусного перетворення типу VII…….……...101
3.4 Висновки до розділу ……103
4 Нові підходи, методи і алгоритми адаптивного кодування зображень ……..105
4.1 Адаптивне використання перетворень ……105
4.1.1 Вибір перетворень в залежності від кореляції вхідних даних ….106
4.1.2 Вибір перетворень в залежності від коефіцієнта стиснення ……106
4.1.3 Вибір перетворень за умови мінімізації обчислень ……...107
4.2 Модифікація оцінок MSE і PSNR…….. ……..……108
4.3 Універсальні алгоритми передбачення значень пікселів ……..108
4.3.1 Універсальний алгоритм передбачення на зовнішніх даних…..109
4.3.2 Універсальний алгоритм інтра-передбачення …….109
4.4 Векторна міра обчислення відстані простору кольорів ……..110
4.4.1 Векторна міра для побудови індексованих палітр кольорів …...111
4.4.2 Векторна міра для оцінки похибки відновлення ……..114
4.5 Індексовані палітри та адаптивний вибір простору кольорів ……115
4.7 Висновки до розділу……...116
Висновки 118
Перелік джерел посилання.....122

Перелік скорочень, умовних познак, одиниць і термінів

ДКП – дискретне косинусне перетворення

ДП – досконалі перестановки

ДІП – двійково-інверсні перестановки

ПАХ – перетворення Адамара – Хаара

ПВК – перетворення високої кореляції

ПНК – перетворення низької кореляції

ПКЛ – перетворення Карунела-Лоєва

ПКГ – перестановки по коду Грея

СХП – слент-Хаара перетворення

СП – слент-перетворення

СКВ – середньоквадратичне відхилення

ЦКП – цілочислове косинусне перетворення

AVIF – AV1 Image Format, формат зображень на основі кодека відео AV1, має режими з втратами і без втрат

FLIF – Free Lossless Image Format, кодек зображень без втрат, використовує алгоритми передбачення значень пікселів і ентропійне кодування

H.265/HEVC – стандарт, формат стиснення відео

HDR – High Dynamic Range, збільшення діапазону яскравості

HEIF, HEIC, BPG – кодеки зображень на основі відеокодека H.265/HEVC

IoT – Internet of Things, інтернет речей

JPEG 2000 – кодек зображень на основі вейвлетів

JPEG XR – формат кодування зображень, розроблений корпорацією Microsoft

JPEG – найбільш популярний растровий формат кодування зображень з втратами, розроблено в 1995 році

JPEG-LS – версія формату JPEG без втрат

Raw – (англ. raw – сирий) –розповсюджений формат даних для нестиснених зображень, отриманих цифровими камерами та сканерами, що не має чіткої специфікації.

RDO – rate-distortion optimization, метод пошуку компромісу між бітрейтом та спотвореннями відео

RGB – адитивна кольорова модель, що кодує кольори за допомогою трьох складових каналів кольорів (Red, Green, Blue)

SAD – сума абсолютних різниць значень пікселів двох зображень чи кадрів

SATD – сума абсолютних різниць коефіцієнтів перетворень

SSIM – structure similarity, метрика порівняння зображень

VP9 – відкритий стандарт кодування відео від корпорації Google

VVC/H.266 – Versatile video coding, стандарт кодування відео

WebP – стандарт компресії зображень з втратами та без втрат на основі відеокодека VP8

YCbCr – кольоровий простір, в якому Y – компонент яскравості, Cb і Cr – компоненти кольоровості, які представляють різницю між синім кольором і яскравістю (Cb) та різницю між червоним кольором і яскравістю (Cr)

QP – параметр квантування

PSNR – пікове співвідношення сигнал/шум

PNG – Portable Network Graphics, формат зображень з втратами та без втрат, оптимальний для інфографіки

PSNR-HVS-M – пікове співвідношення сигнал/шум з урахуванням функції контрастної чутливості (CSF) та міжкоефіцієнтного маскування контрасту DCT

HVS - Human Vision System

ВСТУП

Актуальність. Особливості розвитку області кодування і стиснення зображень та відео полягають в постійному підвищенні кількості і складності методів та алгоритмів, що призводить до значного збільшення обчислювальної складності [1-3]. На заміну фіксованих алгоритмів прийшли адаптивні алгоритми, які відрізняються наявністю декількох алгоритмів чи параметрів та необхідністю пошуку найкращого [4-6]. Це стосується етапів вибору розмірності блоку інтра-прогнозування, порядку перетворення, параметра квантування/стиснення. Оскільки адаптивність значно збільшує обчислювальну складність, це обумовлює пошук, розробку оптимальних комплексів алгоритмів [7-14]. Використання оптимальних наборів алгоритмів має дві форми – по-перше, це пошук оптимальних алгоритмів, які в комплексі найкраще працюють, і по-друге, це порівняння й вибір наборів алгоритмів, що дозволяє зменшити кількість переборів, комбінацій при наступному більш вузькому виборі в рамках обраного набору алгоритмів. Іншими словами, алгоритм вибору найкращого алгоритму розбивається на дві частини, два етапи – вибору комплексу алгоритмів і вибору алгоритму всередині обраного набору алгоритмів. Це є принципово новий підхід кодування та нова архітектура кодека, що включає додатковий рівень – порівняння й вибору наборів алгоритмів, тобто адаптивних режимів [7]. Кодек зображень включає в себе більшу частину алгоритмів кодування відео, тому всі запропоновані підходи, методи та алгоритми повністю стосуються й кодування та стиснення відео, а не тільки кодування зображень. Швидкий розвиток інтернету речей та пріорітет на кодуванні вхідних даних є важливими факторами актуальності пошуку оптимальних та швидкодійних підходів, методів і алгоритмів кодування і стиснення зображень/кадрів з втратами [7-14].

Об'єкт дослідження **–** підходи, методи і алгоритми кодування та стиснення зображень з втратами, комплекси адаптивних та фіксованих алгоритмів.

Мета роботи **–** дослідження та розробка підходів, методів, алгоритмів та адаптивних наборів алгоритмів кодування і стиснення зображень з втратами.

Мета роботи обумовлює наступні завдання: дослідити адаптивні та фіксовані алгоритми в ракурсі підвищення швидкодії; дослідити набори адаптивних алгоритмів; побудувати математичну модель кодування і стиснення зображень; розробити нові адаптивні алгоритми та режими; розвинути нові підходи, методи та алгоритми адаптивного кодування і стиснення зображень з втратами.

Методи дослідження. При розробці й дослідженні нових підходів, методів і алгоритмів кодування та стиснення зображень використовувалися підходи й методи теорії апроксимації, чисельного аналізу, лінійної алгебри, комбінаторики, статистичного й математичного моделювання, кластерного аналізу, теорії інформації.

Наукова новизна отриманих результатів – більшість запропонованих підходів, методів та алгоритмів розглянуто вперше.

Основні отримані результати: математична модель кодування і стиснення зображень з втратами; архітектура кодека, що включає додатковий структурний рівень вибору наборів алгоритмів, адаптивних режимів; концепція кодеків різного рівня, адаптованих до обчислювальних обмежень: швидкодійного, збалансованого, ресурсоємного та універсального зі зворотньою сумісністю; адаптивні режими мінімальної складності; цілочислове просте косинусне перетворення перетворення порядку 16 зі швидкими алгоритмами обчислення без виконання операцій множення; цілочислове однонормове синус-косинусне перетворення типу VII порядку 8 зі швидкими алгоритмами обчислення; новий режим роздільного адаптивного кодування з вибором з трьох перетворень (косинусного, синус-косинусного і синусного); два універсальних алгоритми передбачення значень пікселів на основі зовнішніх (для прогресивної передачі) і внутрішніх даних (з метою зменшення обчислювальної складності) для швидкодійного кодека; алгоритм вибору порядку і типу перетворень в залежності від коефіцієнта стиснення і статистичних характеристик вхідних даних; новий підхід спрощеного адаптивного вибору алгоритмів для всього зображення; алгоритм оцінки статистичних характеристик вхідних даних з метою вибору фіксованих параметрів кодування; модифікація mse, що враховує

лише значення похибки більші за обране X, яка має підвищену точність відображення спотворень, особливо контурів та перепадів значень, а також значно меншу обчислювальну складність; нова векторна міра для обчислення відстані в просторі кольорів, що враховує зміни відтінків кольорів; адаптивні алгоритми вибору простору кольорів: індексованої палітри, RGB та різних режимів YcrCb; нові простори кольорів на основі співвідношення кольорів, нові типи індексованих палітр.

Практичне значення. Отримані результати мають важливе практичне значення і вносять вагомий вклад в розробку та покращення адаптивних кодеків для кожного рівня складності обчислень, в тому числі значно зменшуючи мінімальні вимоги до обчислювальних потужностей. Більшість запропонованих нових підходів, методів і алгоритмів стосуються підвищення швидкодії для швидкодійного та збалансованого кодеків зображень. Запропоновано нові адаптивні режими низької складності та зворотня сумісність кодеків.

Теоретичне значення отриманих результатів складається з декількох складових: 1) запропоновано математичну модель кодування та стиснення зображень/відео з втратами; 2) запропоновано концепцію адаптивних кодеків різного рівня складності і розроблено їх архітектуру з додатковим рівнем вибору адаптивних режимів; 3) запропоновано модифікацію mse і введено нову міру оцінки похибок; 4) запропоновано векторну міру відстані для простору кольорів; 5) запропоновано нові підходи, методи та алгоритми, що дозволяють оптимізувати існуючі кодеки різних поколінь та задають нові напрямки досліджень кодування та стиснення зображень/відео з втратами.

Особистий внесок здобувача. Всі положення й результати дисертаційної роботи, що виносяться на захист, отримано автором самостійно. У роботах, які опубліковані в співавторстві, здобувачеві належать наступні результати: у [8] запропоновано концепцію швидкодіючого кодека зображень; у [15] розроблено швидкий алгоритм обчислення оберненого цілочислового синусного перетворення ЦСП-1 порядку 4 типу VII; у [16] розроблено модифіковане однонормове синус-косинусне перетворення МСКП-1 порядку 8 типу VII і

швидкий алгоритм його обчислення низької обчислювальної складності; у [17] запропоновано модифікований дельта-алгоритм для кодування (інтра-передбачення) зображень, що реалізовано у формі універсальних алгоритмів передбачення значень пікселів поточного блоку на основі зовнішніх даних при використанні прогресивної передачі даних і на основі внутрішніх даних сусідніх блоків зображення.

Апробація результатів дисертації. Результати дисертації доповідались на:

- міжнародній науково-практичній конференції «Інформаційні системи та технології: результати і перспективи», м. Київ, 6 березня 2024 р.;
- міжнародному науковому симпозіуму «Питання оптимізації обчислень (ПОО-XLVIII)», присвяченому 100-річчю від дня народження академіка В.М. Глушкова, м. Львів, 19–22 вересня 2023 р.;
- міжнародній науково-практичній конференції «Інформаційні технології та комп'ютерне моделювання», ІТКМ-2022, 15 – 16 грудня 2022 р., м. Івано-Франківськ;
- X міжнародній науково-практичній конференції "Глушковські читання", м. Київ, 2 грудня 2021 р. ;
- проблемно-науковій міжгалузевій конференції «Юриспруденція та проблеми інформаційного суспільства» (ЮПІС – 2018), 23-25 травня 2018 р., м. Надвірна;
- науковому семінарі "Методи обчислювальної математики" Інституту кібернетики імені В.М. Глушкова НАН України, 26 лютого 2024 р., керівники семінару: академік НАН України В.К. Задірака, академік НАН України О.М. Хіміч.

Публікації. Основні наукові результати дисертаційної роботи викладено в 11 публікаціях, з яких: 2 статті опубліковано в фахових журналах України, 3 статті опубліковано англійською мовою в науковому журналі видавництва Springer та проіндексовано в наукометричній базі SCOPUS, 1 стаття у виданнях України, що не відносяться до категорії фахових за спеціальністю 113 «Прикладна

математика», 5 тез доповідей опубліковано в збірниках доповідей міжнародних та всеукраїнських наукових конференцій та симпозіумів.

Зв'язок роботи із науковими програмами. Дисертація пов’язана із тематикою, що виконувалась і виконується у відділі оптимізації чисельних методів Інституту кібернетики ім. В.М. Глушкова НАН України, результати дисертації знайшли відображення в наступних наукових темах: ВФ 140.18 «Розробити теоретичні основи побудови базисних систем нового класу мультивейвлетів для оброблення сигналів і зображень» № держреєстрації 0119U002275; ВКМ 140.24.21 «Розробити методи і алгоритми розподіленого моделювання та визначення станів віддалених об'єктів тривалого моніторингу на основі суперкомп'ютерних технологій», № держреєстрації 0121U110720; ВК.140.16.15 «Розробка алгоритмів обробки, кодування, аналізу та передачі сенсорних даних для складання трьохвимірних карт місцевості в задачах орієнтації мобільних роботів і безпілотних апаратів»; ВФ 140.14 «Розробити оптимальні за точністю та швидкодією алгоритми розв’язання задач: інтегрування швидкоосцилюючих функцій, цифрової обробки сигналів та зображень, дистанційного моніторингу об’єктів, інформаційної безпеки» № держреєстрації 0114U000357; ВП. 140.19 «Розробити адаптивні методи і алгоритми оброблення та кодування моніторингових даних в безпровідних мережах», № держреєстрації 0119U002273; ВП.140.21 «Розробити ефективні методи та алгоритми оброблення, кодування і аналізу сигналів для дистанційного моніторингу функціональних станів осіб», № держреєстрації 0122U000830.

Структура дисертації.

В **першому розділі** зроблено огляд відомих підходів, методів і алгоритмів кодування зображень та висновки про їх недоліки, актуальність і перспективи.

В **другому розділі** розглядається архітектура адаптивних кодеків різного рівня складності: швидкодіючого, збалансованого, ресурсоємного і універсального. Описана нова математична модель кодування і стиснення зображень з втратами та отримано ряд застосувань у формі нових алгоритмів. Запропоновано новий структурний рівень – порівняння і вибору адаптивних

режимів. Обгрунтовано доцільність зворотньої сумісності кодеків різного рівня і розширення кількості адаптивних режимів для зменшення мінімальних вимог до обчислювальних потужностей. Розглянуто основні характеристики і особливості кожного типу адаптивних кодеків різного рівня складності.

В **третьому розділі** описано нові цілочисельні ортогональні перетворення для адаптивного застосування: просте косинусне порядку 16, модифіковане синус-косинусне типу VII порядку 8 і синусне типу VII порядку 4. Розглянуто умови і доцільність їх застосування, наведено результати тестування і обчислювальна складність.

В **четвертому розділі** дисертації описано нові запропоновані підходи, методи і алгоритми адаптивного кодування і стиснення зображень/кадрів з втратами: новий режим роздільного направленого кодування з адаптивним вибором з трьох перетворень (косинусного, синус-косинусного і синусного); два алгоритми передбачення значень пікселів на основі зовнішніх і внутрішніх даних; новий алгоритм вибору порядку і типу перетворень в залежності від коефіцієнта стиснення і статистичних характеристик вхідних даних; простий алгоритм оцінки статистичних характеристик вхідних даних з метою вибору фіксованих параметрів кодування; модифікація mse, що враховує лише значення похибки більші за обране X; нова векторна, комплексна міра для обчислення відстані в просторі кольорів, що враховує як зміни яскравості, так і зміни відтінків кольорів; адаптивні алгоритми вибору простору кольорів: індексованої палітри, RGB та різних режимів YcrCb; нові простори кольорів на основі співвідношення кольорів, нові типи індексованих палітр.

Структура і об'єм роботи. Дисертація складається із вступу, 4 розділів, загальних висновків, списку літератури зі 94 найменувань. Загальний обсяг дисертаційних досліджень викладено на 131 сторінці друкованого тексту, де обсяг основного тексту – 107 сторінок, зокрема 16 рисунків, 22 таблиці.

РОЗДІЛ 1
ОГЛЯД ВІДОМИХ ПІДХОДІВ, МЕТОДІВ І АЛГОРИТМІВ КОДУВАННЯ ЗОБРАЖЕНЬ

1.1 Огляд відомих форматів кодування зображень

Розглянемо основні найбільш відомі алгоритми і формати кодеків зображень. Найчастіше використовуються наступні графічні формати [4, 5, 18]: JPEG; WebP; PNG; GIF. Також відомі більш складні графічні формати, що створені на базі відеокодеків: формати зображень BPG та HEIF/HEIC мають за основу відеокодек H.265/HEVC [19-21], AV1 Image Format (AVIF) на основі AV1 [22]. Зауважимо, що кодек зображень WebP розроблено на основі VP8 [23].

Окремо виділимо FLIF [24] – новий складний кодек зображень лише на основі передбачення значень пікселів і ентропійного кодування, тобто кодування без втрат, без використання ортогональних перетворень.

Існують і комплексні підходи, наприклад, DjVu [4, 5, 18], де використовуються такі алгоритми: алгоритм відділення тексту від фону; вейвлетний алгоритм кодування фону IW44; алгоритм стиснення чорно-білого зображення JB2; універсальний алгоритм стиснення ZP; алгоритм часткового відкриття за запитом; алгоритм виділення шарів і масок (фон, передній план та чорно-біла маска). До кожного із шарів застосовуються окремі алгоритми.

Комплексний підхід формата DjVu можна перенести в концепцію створення нових графічних форматів. З іншого боку, складний і комплексний стандарт MNG, який включає в собі два формати – PNG та JPEG [25], та ще й функцію анімації, не отримав широкого розповсюдження.

Основними алгоритмами стиснення зображень з втратами є: блочні перетворення; вейвлет-перетворення; фрактальні перетворення; проріджування і квантування даних. Найбільш поширені блочні перетворення [26-30], вони відрізняються високою якістю відновлених даних і низькою складністю обчислень. Але блочні перетворення мають і недоліки, зокрема наявність

блочних артефактів при середніх та великих ступенях стиснення. Вейвлет-перетворення також мають ряд недоліків: висока обчислювальна складність і менший коефіцієнт стиснення, особливо при малих і середніх коефіцієнтах стиснення, оскільки попередньо до зображень не застосовуються алгоритми передбачення значень пікселів, які дають додаткове стиснення без втрат. Алгоритми передбачення значно спотворюють структурні закономірності вхідного зображення, тому застосування їх одночасно з вейвлетами недоцільно, оскільки вейвлети мають ієрархічну структуру для відображення вхідних даних у різних масштабах, яка у випадку значної хаотичності початкових даних буде недостатньо ефективною. Хоча теоретично цікаво було б дослідити кодек зображень на основі вейвлет-перетворень, але з використанням алгоритмів передбачення значень пікселів. Алгоритми фрактальних перетворень мають значні коливання якості для різних видів зображень, тому не набули поширення.

Цікавою та перспективною є ідея побудови мультивейвлетів фрактального типу, які мають переваги перед класичними – самоподібність дозволяє використовувати прості регулярні алгоритми для побудови мультивейвлетів наступного рівня. В [31-37] запропоновано і досліджено мультивейвлети фрактального типу, які мають швидкі алгоритми обчислення дуже низької обчислювальної складності та вищу якість за відомі класичні мультивейвлети, і можуть ефективно використовуватись для аналізу зображень.

Для кодування зображень без втрат якості застосовуються такі алгоритми: алгоритми передбачення (коефіцієнтів перетворення або значень пікселів); алгоритми ентропійного кодування. Блочні перетворення і вейвлети (мультивейвлети) без квантування коефіцієнтів формують так званий режим «без втрат», який має мінімальні втрати інформації в результаті заокруглення коефіцієнтів до цілих значень та незначної похибки апроксимації.

Виділяють векторні і растрові формати (хоча комплексний підхід також є можливим) [4, 5]: векторні (CDR, EPS, WMF, AI та інші), растрові (TIFF, PNG, GIF, BMP, JPEG, PSD та інш.). Векторний формат представляє собою набір геометричних примітивів. Більшість векторних форматів можуть

використовувати і растрові дані, або ж посилання на них (технологія OPI), і є фактично гібридними. Растровий графічний файл представляє собою таблицю, «карту бітів» (bitmap). BMP – найпростіший растровий формат, без втрат і без стиснення. Йому на заміну можна розробити більш сучасний формат для збереження даних без втрат, спрощений аналог FLIF.

Представляє цікавість формат GIF [4, 5] (Graphics Interchange Format), який використовує лише 8 бітів для передачі обмеженої палітри кольорів і тому ефективно стискає фон, зберігаючи основні деталі зображення. Також GIF підтримує анімацію і альфа-канал прозорості. Концепція використання обмеженої та фіксованої індексованої палітри кольорів (у моделі кольорів RGB), в першу чергу для окремих типів зображень/кадрів (чи областей зображень) залишається актуальною і перспективною, незважаючи на її мале поширення в сучасних кодеках. Додаткові кольори, що є відсутніми в базовій палітрі GIF, можна отримати за допомогою алгоритмів розсіювання, генерації шуму та накладення періодичної структури (dithering, noise та pattern) [4, 5].

JPEG (Joint Photographic Experts Group) [25] є найбільш відомим форматом стиснення із втратами, хоча він не підтримує альфа-канал (прозорість) та анімацію, незважаючи на те, що він був розроблений після GIF, і міг включити в себе всі його особливості в якості складової частини (включаючи адаптивне застосування індексованої палітри кольорів). JPEG використовує растрові дані, які зберігаються в моделях кольорів RGB, YCbCr та відтінки сірого.

На зміну JPEG було розроблено ряд нових форматів, в тому числе зі зворотньою сумісністю, зокрема JPEG XR, JPEG XS, JPEG XL. JPEG XR (відомий також як HD Photo або Windows Media Photo) забезпечує покращення стиснення і підтримує більш високі бітові глибини (до 32 біт на канал), альфа-канали та HDR-зображення.

JPEG XS є новітнім стандартом, створеним для швидкої передачі зображень із низькою затримкою для застосувань, де важлива швидкість передачі без значних втрат якості. Хоча JPEG XS забезпечує стиснення з втратами, але з високою якістю зображення.

JPEG XL є універсальним форматом, призначеним для заміни як JPEG, так і інших форматів зображень, включаючи PNG, WebP і навіть TIFF. Він підтримує як стиснення з втратами, так і без втрат. Особливості JPEG XL включають високу ефективність стиснення, швидке декодування, підтримку великих зображень (до мільярда пікселів), велику бітову глибину (до 32 біт на канал), а також підтримку альфа-каналів і HDR-зображень. Схему роботи JPEG XL наведено на рис. 1.

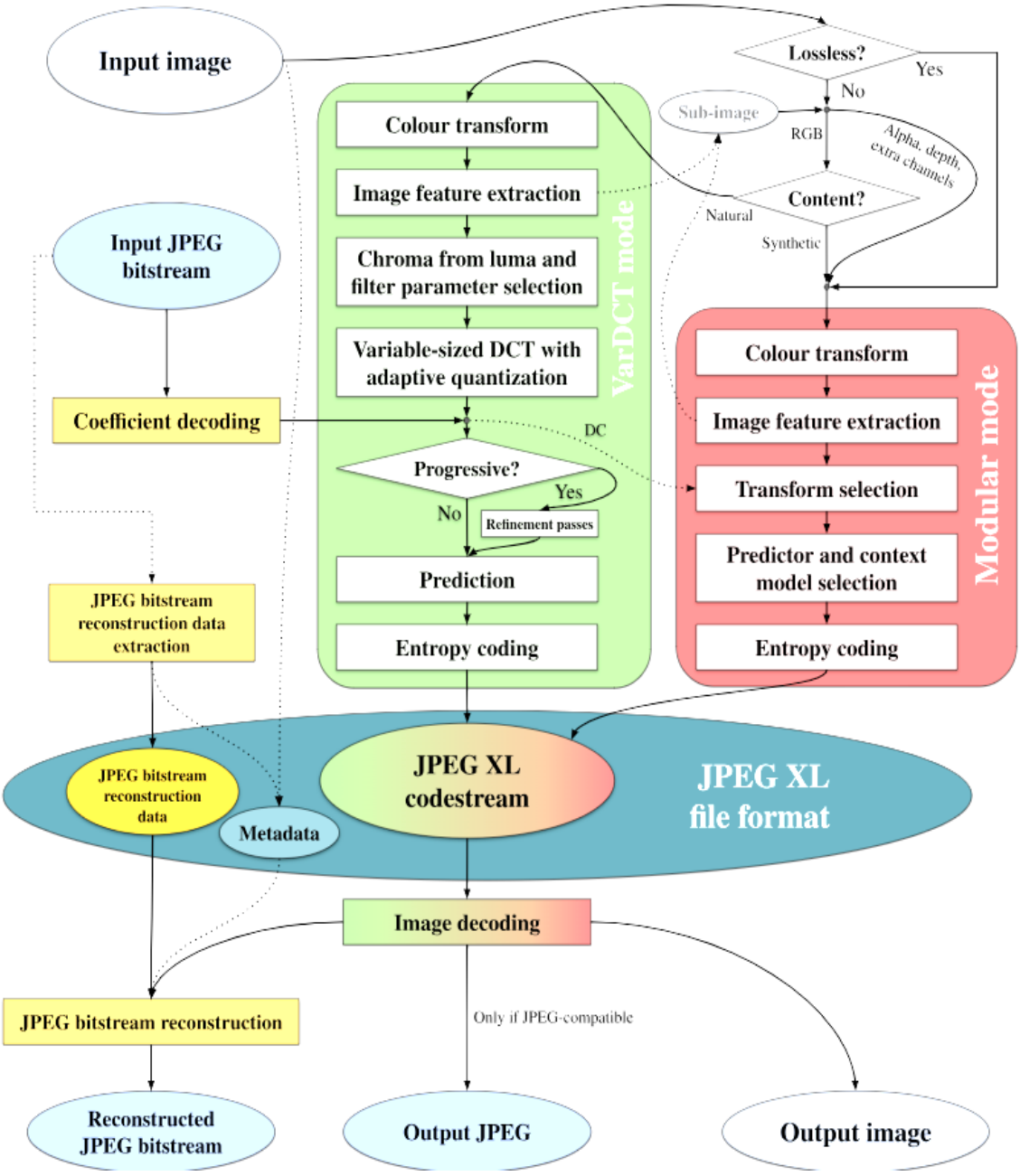

Рисунок 1. Схема роботи JPEG XL

PNG (portable network graphics) мав замінити формат GIF, добре підходить для чорно-білих зображень (текстових, графіків тощо) і має реалізацію альфа-каналу. Формально формат PNG без втрат якості, але застосування прорідження даних обумовлює втрати інформації, і в цілому для зображень з багатою гамою кольорів не дуже підходить, тому повнокольорові зображення у форматі PNG мають більший розмір файлів, ніж у інших кодеків. PNG має дві реалізації: PNG8, що використовує індексовану палітру кольорів, і PNG24 з 24 бітним простором кольорів.

Теоретично PNG міг легко реалізувати адаптивне використання обох підходів на програмному рівні, розробивши простий алгоритм вибору (обчислення кількості кольорів) і критерій адаптивного вибору (тобто кількість кольорів вище певного заданого чи обраного значення P обумовлює формат PNG24, нижче – PNG8), але це не було запропоновано, залишивши вибір між форматами PNG24 і PNG8 на розсуд користувачів, більшість з яких цим питанням не переймались. В результаті більша частина інфографіки з обмеженою кількістю кольорів кодується у форматі PNG24, а не PNG8.

1.2 Дискретні ортогональні перетворення

Квантування коефіцієнтів перетворення є основним алгоритмом отримання великих коефіцієнтів стиснення з втратами. Відповідно, важливою складовою кодеків зображень/відео на основі блочних перетворень є вибір ортонормованої системи базисних функцій перетворення, які мать різну ефективність апроксимації вхідних даних.

В роботах [15,16, 38-51] наведено різноманітні блочні перетворення, які відрізняються порядком і типом, порядку 4, 8, 16 та 32, тип перетворення – косинусні, прості косинусні, ступінчасті косинусні, синусні типу VII, модифіковані синусні, синус-косинусні, косинус-синусні та інші (і це лише частина всіх існуючих блочних перетворень).

Хоча косинусні перетворення в цілому мають найкращу ефективність ущільнення інформації (часто пишуть енергії) на один коефіцієнт [4, 6], але за необхідності підвищення швидкодії можуть використовуватись такі перетворення, як Хаара, Уолша-Хаара, слент-перетворення та інші спрощені перетворення, особливо при умові попереднього використання алгоритмів стиснення без втрат, тобто алгоритмів передбачення значень пікселів, які значно зменшують різницю в ефективності кодування перетворень різного класу. Але оскільки алгоритми передбачення значень пікселів були розроблені після 2002 року [6], коли популярність спрощених перетворень почала зменшуватись внаслідок підвищення обчислювальних потужностей техніки, така комбінація не отримала поширення, незважаючи на її простоту і очевидність.

З метою підвищення швидкодії замість дискретних перетворень, що використовують числа з плаваючою комою, часто використовуються цілочисельні апроксимації, які можна називати цілочисельними перетвореннями, але чомусь ця назва не набула поширення, і досить часто продовжують писати наприклад ДКП, хоча мова йде про ЦКП (цілочисельне косинусне перетворення).

Ортогональність ЦКП досягти складніше, ніж ДКП. Але дослідники помітили, що незначна похибка ортогональності нівелюється похибкою квантування коефіцієнтів перетворення, що розширило можливості пошуку, підбору цілочисельних значень матриць перетворень.

З метою подальшого зменшення обчислювальної складності були розроблені однонормові цілочисельні перетворення, що зменшує обчислювальну складність нормалізації коефіцієнтів перетворення, а для зменшення кількості множень почали використовувати числа степені двійки [26-29], адже в такому випадку ділення при нормуванні можна замінити на зсуви.

Розробка швидких алгоритмів обчислення перетворень на основі факторизації матриць з одночасним зменшенням кількості операцій множення є одним з напрямків зменшення обчислювальної складності перетворень [38-45].

1.3 Алгоритми передбачення значень пікселів

Розробка і застосування алгоритмів передбачення значень пікселів поточного блоку на основі значень сусідніх блоків зображення/кадру стало важливим кроком до збільшення ефективності кодування і стиснення. Як вже було сказано, вперше вони були застосовані в 2003 році в main профілі кодека H.264 [6], причому одразу для двох розмірів блоку, 4 і 16, між якими робився адаптивний вибір. Застосування розміру блоку передбачення на 16 обумовлено зменшенням обчислювальної складності, адже для кожного розміру блоку передбачення адаптивно обирається один алгоритм (до того ж для блоку 16x16 вибір робився з 4 алгоритмів, а не з 9, як для блоку розміру 4).

Крім відомих підходів і алгоритмів, слід очікувати появу нових способів передбачення значень пікселів, з урахуванням геометричних форм, а не блоків, які є штучним утворенням, на відміну від природнього розподілу контурів і областей зображення. Одним з напрямків цього підходу є використання нейромереж відповідного типу.

Масштабування при прогресивній передачі даних, наприклад, в кодеку LCEVC [52] створює новий тип передбачення значень пікселів – на основі значень пікселів зображень/кадрів меншого масштабу, тобто на основі зовнішніх даних.

Такий тип передбачення характеризується меншою обчислювальною складністю, оскільки він універсальний (тобто значення пікселя меншого зображення є середнім значенням блоку пікселей більшого зображення), і не потребує адаптивного порівняння й вибору з декількох алгоритмів передбачення.

В той же час збільшення кількості алгоритмів передбачення з кожним наступним поколінням кодеків (від 10 у WebP, 9 у H.264 до 35 у H.265 та 67 у H.266) дозволяє краще стискати вхідні дані без втрат, хоча й значно зменшує ефективність використання блочних перетворень (а також дуже збільшує мінімальні вимоги до обчислювальної потужності).

1.4 Адаптивні алгоритми кодування зображень

На заміну фіксованим параметрам і алгоритмам прийшли адаптивні, що дозволило значно збільшити ефективність стиснення, але й відповідно збільшило складність обчислень.

Адаптивні алгоритми включають в себе алгоритм чи алгоритми порівняння та вибору одного з декількох можливих алгоритмів чи параметрів. В найскладнішому варіанті доводиться узгоджувати між собою декілька різних алгоритмів порівняння.

Розробка й порівняння адаптивних алгоритмів є одним з важливих напрямків досліджень, що обумовлюється різноманіттям критеріїв розробки й вибору адаптивних алгоритмів – збільшення коефіцієнта стиснення, збереження високої інформативності (особливо актуально для медичних, космічних та інших спеціалізованих даних) та підвищенням швидкодії, зменшенням обчислювальної складності, подовженням терміну роботи від акумулятора, що актуально для інтернету речей, носимої електроніки та інших застосувань [53-60]

Для ілюстрації обчислювальної складності вибору алгоритмів на рис. 2 показано розбиття на блоки розміру від 4 до 64 для етапів передбачення і перетворення кодеків H.265 та HEIF. Як відомо, розмір блоку передбачення може бути більшим за розмір матриці перетворення, але не може бути меншим, за виключенням використання універсального алгоритму на зовнішніх даних при прогресивній передачі даних – адже в цьому випадку не потрібно декодувати сусідні блоки для застосування алгоритмів передбачення поточного блоку на основі відновлених значень сусідніх блоків.

Адаптивний вибір параметра квантування коефіцієнтів також значно ускладнив процес кодування, на додачу до вибору розміру блоку передбачення, конкретного алгоритму передбачення для обраного блоку і порядку перетворення. Враховуючи невизначеність та діапазон оцінки якості зображення, незначне збільшення або зменшення параметру квантування відносно оптимального (за певними критеріями) є допустимим.

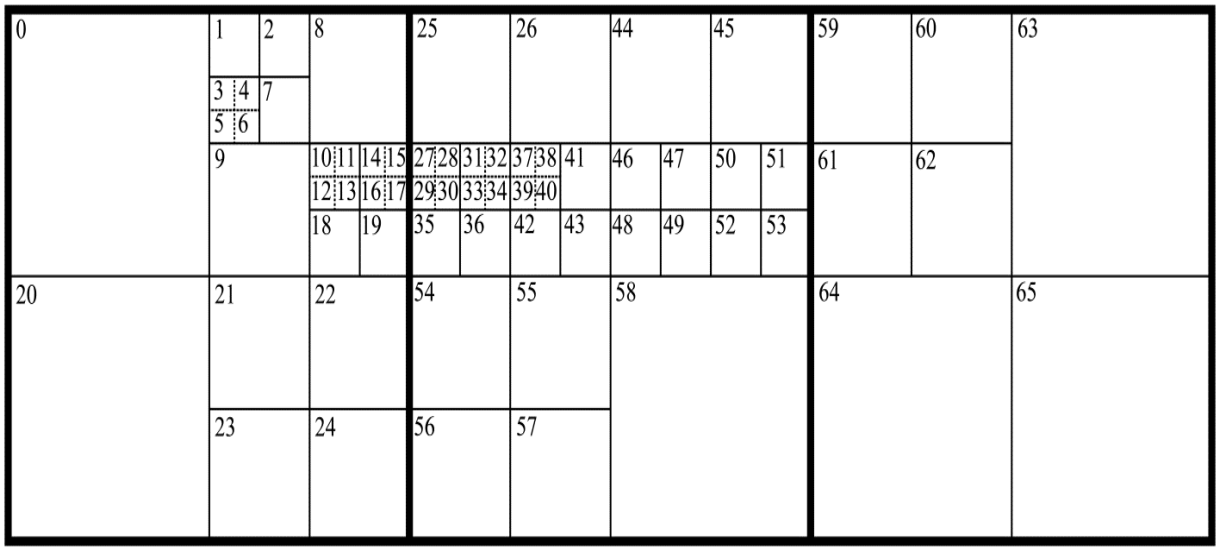

Рисунок 2 Розбиття на блоки розмірності від 4 до 64 на етапах передбачення і вибору перетворення у кодеках H.265 та HEIF

Іншими словами, якість зображення/кадру можна стабільно перевищувати, якщо дозволяють умови, або навпаки, зменшувати при необхідності – результат буде попадати в діапазон прийнятної візуальної якості, а витрати на пошук оптимального параметру квантування значно зменшаться.

Ще одним фактором збільшення обчислювальної складності адаптивних алгоритмів є розробка та застосування роздільних адаптивних перетворень, які потребують експериментального перебору всіх можливих комбінацій перетворень, що входять до них, зазвичай косинусних і синусних перетворень.

1.5 Співвідношення похибки до коефіцієнта стиснення і критерії якості зображень

Коефіцієнт стиснення вхідних даних залежить від їх статистичних характеристик, ефективності алгоритмів стиснення без втрат (ентропійних та передбачення значень пікселів), але найбільшу роль відіграє параметр квантування коефіцієнтів перетворення. В результаті ділення коефіцієнтів перетворення на параметр квантування (зазвичай з заокругленням до цілих значень) відкидається певна частина інформації, в першу чергу –

високочастотної, на другому місці за рівнем спотворень є низькочастотні дані, і найменше страждають в результаті апроксимації середні частоти.

Адаптивний вибір параметра квантування робиться в рамках набору алгоритмів RDO (Rate-distortion optimization) [61-64]. Відповідно, при обмеженнях на бітрейт (швидкість передачі даних) для відео або обмеженнях на коефіцієнт стиснення для зображень рівень похибок є другорядним фактором вибору, і навпаки – при відсутності обмежень рівень похибок є визначальним критерієм вибору, підбору параметра квантування. Задача адаптивного вибору параметра квантування є складною і має різні підходи та критерії [65-68].

Важливим фактором вибору параметру квантування є критерії якості відновлених зображень/кадрів. З плином часу ці критерії постійно підвищуються та змінюються. Якщо недавно було достатньо 8 біт на канал кольору (а часто це було лише 5-6 біт – з урахуванням значного квантування коефіцієнтів перетворення точність падала до 56 біт на канал, до того ж навіть 8 бітні дані відображались в основному на найбільш розповсюджених 6 бітних TFT екранах), то очікується збільшення точності кольорів до 10-12 біт на канал RGB (що знову ж таки буде доступно в першу чергу для невеликої частини якісних екранів). Аналогічно спостерігаються процеси постійного зростання роздільної здатності, від HD до full HD з перспективою 4К та навіть 8К чи 16К.

Зміни критеріїв якості впливають на параметри квантування різних частот. Якщо раніше пріорітетними були низькі та середні частоти (що проілюстровано на рис. 3 на прикладі матриці квантування коефіцієнтів перетворення в JPEG), то в стандарті H.265 вже застосували один параметр квантування до всіх коефіцієнтів перетворення [26], тобто збереження високочастотної інформації стало більш важливою задачею.

Хоча деякі критерії, що намагаються врахувати особливості візуального сприйняття людини (HVS) частково використовують підхід пріорітета низько- і середньочастотної інформації, на додачу до оцінок чіткості контурів.

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

Рисунок 3 Матриця квантування коефіцієнтів перетворення JPEG

Підвищення якості зображень/відео може торкнутись і змін простору кольорів, що зменшить частоту використання спрощених режимів YCbCr і збільшить поширення RGB або інших просторів кольорів.

1.6 Роздільні адаптивні перетворення

Роздільні адаптивні перетворення [69] потребують емпіричного перебору всіх можливих комбінацій косинусних і синусних перетворень певного порядку, що значно збільшує обчислювальну складність, тобто замість 2D косинусного перетворення використовують різні комбінації 1D косинусного і синусного перетворень для рядків і стовпців (що додатково потребує збереження службової інформації про конкретну комбінацію).

Але обмеженість доступних варіантів збільшення коефіцієнту стиснення і пріорітет на збільшенні стиснення стали причинами для подальшого розвитку даного підходу в кодеку H.266 (VVC) [70] .

На додачу до роздільних адаптивних перетворень порядку 4, що використовувались в кодеку H.265, для H.266 були розроблені роздільні адаптивні перетворення вищих порядків.

Які є більш економічними з точки зору кількості службової інформації, тобто співвідношення об’єму службової інформації щодо конкретної обраної комбінації до об’єму стиснутих даних.

1.7 Прогресивна передача даних

Прогресивна передача даних є важливою особливістю кодеків відео для адаптації до умов передачі та відображення. Але для зображень прогресивна передача також є актуальною і необхідною, оскільки вона зменшує обчислювальну складність кодування, якщо розділяти зображення на декілька пов’язаних зображень різних масштабів і кодувати різниці між ними.

Також прогресивна передача даних є адаптивним підходом збільшення коефіцієнту стиснення. Іншими словами, зображення меншого масштабу має вищу інформаційну щільність в результаті зменшення масштабу, а виникаючі різниці стискаються адаптивно, з використанням критеріїв HVS, тобто частина різниць відкидається і не кодується, оскільки вони не впливають на візуальну якість зображення, будучи менше певного порогового значення.

Відповідно, зменшення масштабу є алгоритмом простого у виконанні додаткового стиснення (не для всіх областей зображень, тому він є адаптивним алгоритмом).

Частково подібний підхід використано в WebP і VP8, де використовується алгоритм масштабування відновлених даних з подальшою фільтрацією. Якби розробники WebP додатково кодували найбільш важливі дані для покращення збільшеного зображення (а це в основному контури й значні перепади), вони б могли відкрити цей простий адаптивний алгоритм стиснення.

Прогресивна передача даних також дає можливість отримати більш точну інформацію за запитом. Для останього варіанту необхідно розробити окремий протокол, що пов’язує файли різної якості, масштабу, для встановлення каналу зв’язку за запитом.

Ймовірно, це може бути реалізовано в рамках використання інтелектуального стиснення за допомогою нейромереж. Розглянемо більш детально питання й напрямки використання нейромереж.

1.8 Перспективи застосування нейромережевих методів для стиснення і відновлення зображень/кадрів з втратами

Використання нейромереж для кодування і стиснення відео розглядалось перед початком розробки кодека H.266. Значна обчислювальна складність цього підходу змусила відкласти його застосування до наступного покоління. Зараз ведеться розробка кодека H.267 на основі нейромереж, і очікується, що в 2025 році він буде розроблений. Розглянемо які підходи можуть бути застосовані в ньому.

1.8.1 Застосування нейромереж для розробки нових підходів, методів і алгоритмів стиснення і відновлення зображень/кадрів

Використання нейромереж для кодування і стиснення з втратами зображень і відео є підходом, що може дати значне підвищення ефективності стиснення [71, 72]. Як і раніше, чим більше ефективність стиснення – тим більша обчислювальна складність алгоритмів, і це стосується й використання нейромереж.

Процес застосування нейромереж складається з двох частин: кодування, пошуку набору параметрів, що ефективно і з обраною точністю відображають вхідні дані, і декодування, що включає в себе й етап відновлення, аналогічний етапу фільтрації у звичайних кодеках.

Використовуючи додаткову інформацію, отриману при навчанні, нейромережі здатні суттєво покращити отримані при декодуванні дані. Відповідно, іншим напрямком застосування нейромереж є наближене відновлення неповних даних, що передаються звичайними кодеками, тобто покращена фільтрація. Ця задача має меншу обчислювальну складність порівняно з задачею кодування й декодування за допомогою нейромереж. Третім напрямком використання нейромереж є застовування їх для адаптивного вибору існуючих методів і алгоритмів для їх оптимізації.

1.8.2 Застосування нейромереж для наближеного відновлення початкових даних

Часткове використання нейромереж лише для наближеного відновлення, фільтрації отриманих даних може використовуватись одночасно і паралельно з існуючими звичайними кодеками [73, 74]. Але навіть цей варіант, що потребує значно менше обчислювальних потужностей, вимагає ряд додаткових умов для широкого розповсюдження.

Одним з випадків подібного застосування нейромереж є різноманітні фільтри, які можуть як фільтрувати дані, так і додавати нові особливості, згідно налаштуванням користувачів. Наприклад, можна робити карікатури або мультиплікаційні зображення, або інші варіації (наведемо приклад для ілюстрації можливостей нейромережі на рис.4), які мають менший об’єм відносно оригінальної інформації.

Зауважимо, що в цілому підхід наближеного відновлення даних має принципову похибку – адже метою є покращення візуальної якості, а не підвищення точності відновлення.

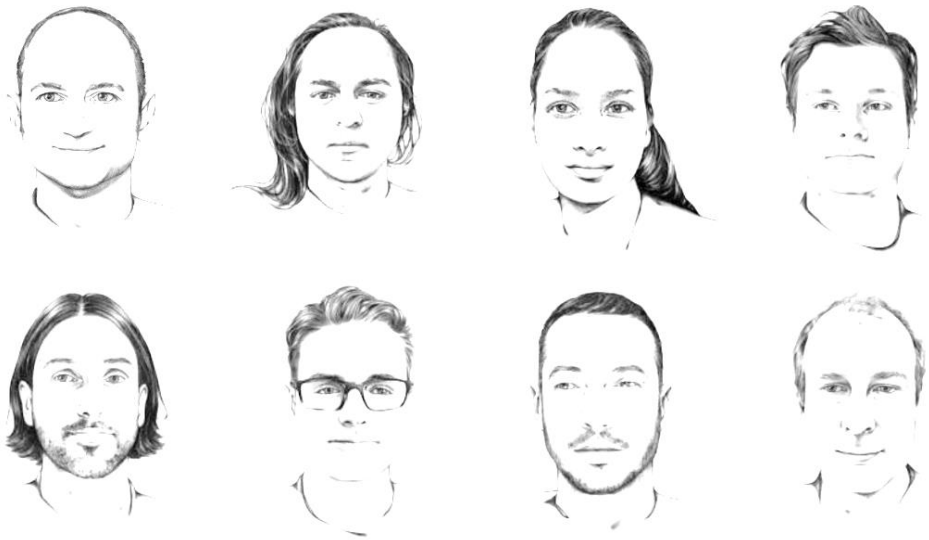

Рисунок 4 Приклад використання нейромережі для обробки фото

1.8.3 Застосування нейромереж для оптимізації адаптивного використання алгоритмів стиснення і фільтрації зображень

Використання нейромереж для оптимізації використання вже існуючих алгоритмів і кодеків є третім напрямком досліджень. Нейромережі можуть робити складний інтелектуальний пошук оптимальних локальних алгоритмів і параметрів кодування [75].

Разом з попереднім підходом застосування нейромереж для наближеного відновлення ці два підходи можуть бути застосовані вже на сучасному етапі розвитку нейромереж, причому із прийнятною обчислювальною складністю, на відміну від першого підходу (розробки нових підходів, методів і алгоритмів засобами нейромереж), який потребує як розвитку нейромереж, так і більшої обчислювальної складності відносно інших підходів.

1.9 Висновки до розділу

В першому розділі розглянуто особливості існуючих форматів і алгоритмів кодування зображень. Наявність недоліків у кожному з відомих кодеків обумовлює потребу у розробці нових підходів. Перерахуємо основні проблеми кожного з існуючих форматів.

FLIF, PNG – для режиму компресії з втратами застосовують проріджування, а не перетворення, що не є оптимальним, оскільки проріджування це просто відкидання, втрата частини інформації. Квантування коефіцієнтів перетворення значно ефективніше для збільшення стиснення.

HEIF/HEIC, BPG і AVIF – не використовують прогресивної передачі даних і мають занадто високу обчислювальну складність, яка мало змінюється (нема адаптивних режимів низької складності для певних умов використання).

JPEG – не має алгоритмів передбачення значень пікселів, хоча це один із основних алгоритмів стиснення без втрат. Використовує частковий замінник – простий алгоритм передбачення DC-коефіцієнтів перетворення. Також

використовує зменшення значень пікселів на 128 – що по суті є одним із найбільш простих методів передбачення (це середнє значення діапазону від нуля до 256). Обчислювальна складність більша за рахунок використання ДКП замість швидких алгоритмів обчислення ЦКП. Має прогресивну передачу даних, але неефективну. Використовує завеликі значення параметрів квантування для низьких частот, що погіршує візуальну якість.

WebP використовує ДКП порядку 4, яке неоптимальне за критерієм коефіцієнта стиснення, і має більшу обчислювальну складність відносно сучасних цілочисельних косинусних перетворень зі швидкими алгоритмами. Має занадто низькі вимоги до якості зображень, зокрема використовуючи проріджения у форматі 4:2:0, не має адаптивних режимів високої якості. Також не реалізовано прогресивної передачі даних.

Новий графічний формат має використовувати ЦКП зі швидкими алгоритмами, мати якісну реалізацію прогресивної передачі даних, використовувати алгоритми передбачення значень пікселів (адаптивно змінюючи їх кількість) і мати адаптивні режими для різних типів зображень та адаптивні режими різної складності.

Серед алгоритмів були розглянуті блочні перетворення, вейвлети і мультивейвлети, в тому числі – зі швидкими алгоритмами обчислення та фрактальною природою (самоподібністю).

Також був проаналізований комплексний підхід, застосований у форматі DjVu, тобто виділення масок, слоїв зображення (передній план, фон та чорно-біла маска).

Була приділена увага перспективам застосування нейронних мереж, фіксованим палітрам кольорів, векторній графіці та її комплексному застосуванню разом з растровою та алгоритмами масштабування.

РОЗДІЛ 2
НОВА АРХІТЕКТУРА АДАПТИВНОГО КОДУВАННЯ І СТИСНЕННЯ ЗОБРАЖЕНЬ

Запропонована архітектура адаптивного кодування і стиснення зображень (а також відео) відрізняється від існуючої наступними особливостями:

- додаванням ще одного структурного рівня, на якому порівнюються та обираються адаптивні режими, тобто набори алгоритмів;
- адаптивності до обмежень обчислювальної потужності – для кожного рівня обчислювальних ресурсів пропонується свій оптимальний кодек;
- використанням нових підходів, методів і алгоритмів;
- адаптивним застосуванням відомих алгоритмів (індексована палітра);
- застосуванням нових мір обчислення похибок.

В рамках запропонованої архітектури розглянемо наступні адаптивні кодеки за співвідношенням швидкодії і стиснення:

1) швидкодійний кодек – з пріорітетом на швидкодії;
2) збалансований кодек – з урахуванням і швидкодії, і стиснення;
3) ресурсоємний кодек – з пріорітетом на стисненні (сучасний підхід);
4) універсальний кодек – інтегральний кодек, об’єднання всіх трьох типів кодеків в одному з адаптивним вибором режимів.

2.1 Основні поняття і закономірності

Розглянемо базові поняття та закономірності, що важливі для подальших досліджень та формування і розуміння математичної моделі.

Апроксимація даних – основний процес в кодуванні та стисненні зображень з втратами. Починається вона з апроксимації аналогових вхідних даних цифровими даними (рис. 5).

Початкові проквантовані цифрові дані апроксимуються в просторі вибраних базисних функцій обмеженої кількості і точності.

Квантування коефіцієнтів перетворення дозволяє значно зменшити об’єм отриманої інформації за рахунок відкидання частини інформації. Квантовані коефіцієнти представляють собою апроксимацію початкових коефіцієнтів.

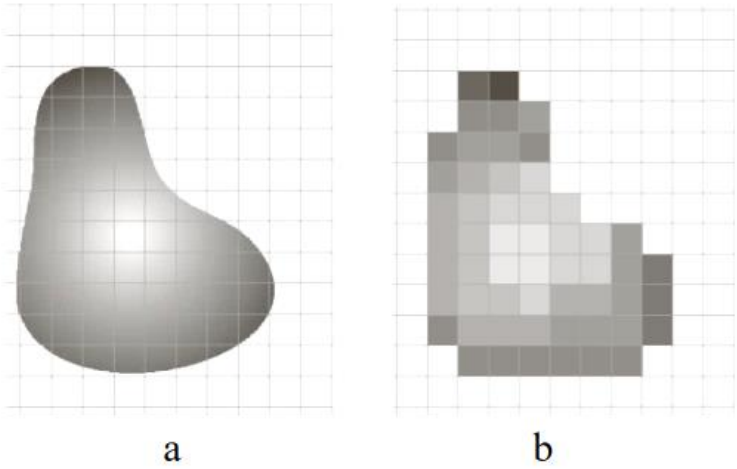

Рисунок 5. Формування (квантування) цифрових даних на основі аналогових

Зображення є проекцією трьохвимірного простору на двовимірний, тобто відображає структурованість, наявність сукупності окремих об’єктів (рис. 6). При цьому спостерігається надлишковість графічних даних, оскільки існують структурні закономірності, взаємозв’язаність, кореляція між значеннями точок, пікселів відображення окремого об’єкта. Ця надлишковість дозволяє отримувати стиск без втрат (або з мінімальними втратами) при кодуванні зображень.

В разі значного стиснення втрати початкової інформації можуть бути помітними, у вигляді зміни кольорів, неспівпадіння кольорів та відтінків кольору, розмитості перепадів, границь об’єктів і так званих блочних артефактів (які з’являються внаслідок використання блочних перетворень і різноманіття спотворень для різних блоків зображення).

При відображенні трьохвимірного простору в двовимірний виникають спотворення, як структурні, так і хаотичні (різні типи шумів, які характеризуються конкретними параметрами розподілу випадкового процесу – імпульсний, гаусів, низько- або верхньочастотний та інш.).

Кодування і стиснення зображень з втратами також вносить два типи спотворень – структурні, закономірні та випадкові шуми.

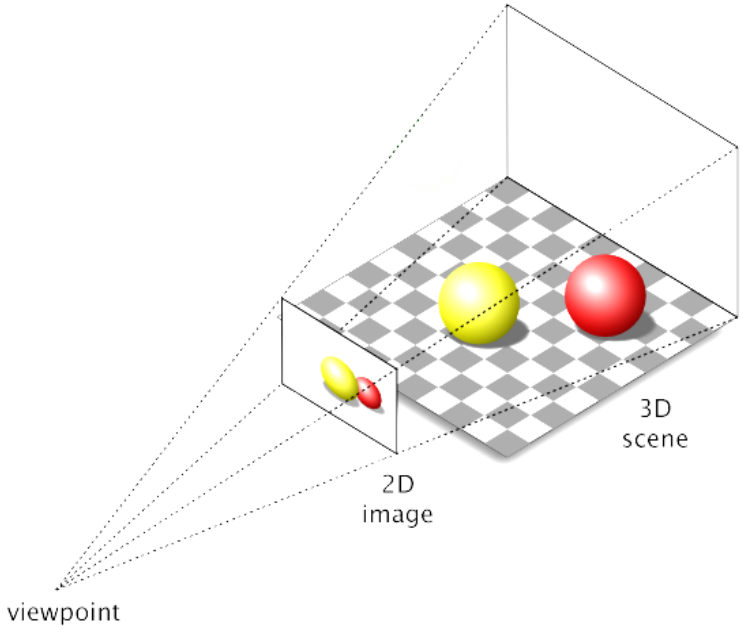

Рисунок 6. Ілюстрація двовимірних зображень як проекції трьох вимірів

Важливим параметром є масштаб відображення, або роздільна здатність. Тобто одна і та ж графічна інформація, зображення може мати різні масштаби, розміри, точність.

При обробці і кодуванні зображення виникає локальний масштаб – розмір блоку даних, що обробляється. Він суттєво впливає на результат кодування – чим менший блок даних обробляється перетворенням, тим вища точність апроксимації випадкових або складно структурованих даних.

Саме тому для кодека H.264 для main профіля було обрано перетворення порядку 4, а не 8, як для кодека попереднього покоління H.263, і в більш складному high профілі використовується не тільки перетворення порядку 8 (відкриття швидких алгоритмів для його обчислення стало основною причиною розробки цього нового профіля), а й перетворення порядку 4, незважаючи на менший коефіцієнт стиснення.

2.2 Математична модель кодування і стиснення зображень

Розглянемо та сформулюємо нову математичну модель кодування і стиснення зображень з втратами.

Двовимірні дані, що мають виміри *x* та *y*, часто представляють у формі функції двох зміних $f(x,y)$.

Для першого наближення цього достатньо, хоча більш точно представляти дані як суму ряда функцій, а для відображення розривів (переходів від відображення одних об'єктів до інших) можна представляти як об'єднання функцій (сум функцій), що визначені на різних інтервалах.

Крім того, потрібно враховувати шум (хаотичні, випадкові зміни, коливання значень) як складову частину, на додаток до закономірностей, функціональних залежностей.

Причому всі функції можна розділити на дві групи: перша група ефективно апроксимується в базисі обраних функцій (наприклад, фур'є, косинусних, вейвлет-функцій, функцій Хаара чи Уолша і т.д.), а друга група має складні залежності, що неефективно передаються, апроксимуються блоком обраних базисних функцій.

З урахуванням вищезазначеного, замість спрощеного представлення вхідних даних у формі $f(x,y)$ краще розглядати більш складну і точну форму представлення:

$$f(x,y)=\sum_i g_i(x,y)+R(x,y)+P,\ x,y\in s_i,$$

де s_i - інтервали, на яких визначені функції $g_i(x,y)$, $R(x,y)$ – складна функція (сума функцій), що погано піддаються апроксимації (як і *P*), *P* – хаотична шумова складова.

Теорема Шеннона стверджує, що існує теоретичний ліміт стиснення інформації без втрат, який залежить від рівня ентропії початкових даних.

Іншими словами, для кожного рівня ентропії є свій теоретично можливий ліміт стиснення без втрат. Тобто, чим більший рівень ентропії початкових даних – тим менше можна їх стиснути без втрат.

Аналогічна теорема Шеннона для джерела без пам’яті обгрунтовує теоретичний ліміт стиснення інформації з втратами, який також залежить від рівня ентропії вхідних даних (тобто чим вища ентропія – тим менше стиск).

В нашому випадку $R(x,y)$ + P є складовими ентропії E, яка обмежує можливість стиснення даних.

Але треба враховувати додаткові критерії вибору коефіцієнта стиснення, а саме, що крім обраного, заданого рівня похибок (запланованих втрат інформації) має вплив обмеження на об’єм даних. Якщо потрібно стиснути більше – то втрати інформації збільшують, сильно квантовані коефіцієнти перетворень формують більш грубу, наближену апроксимацію вхідних даних.

З іншої сторони, навіть повністю хаотичні дані (наприклад, штучно згенерований шум, рис. 7) наближено представляються апроксимацією в базисі вибраних ортогональних функцій (можна сказати – представляються рядом послідовних апроксимацій, для кожного параметра квантування).

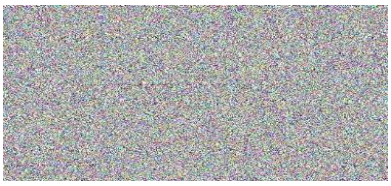

Рисунок 7. Штучно згенерований випадковий шум

Причому, як вже було вказано вище, чим менше порядок перетворення – тим точніше апроксимуються хаотичні та складно структуровані дані.

Одна з основних закономірностей математичної моделі стосується залежності від ентропії E (що розкладається на дві складових, інформаційну $R(x,y)$ і хаотичну P), що обумовлює теоретичне обмеження на коефіцієнт стиснення для будь-якого обраного значення кількості відкинутої інформації, яка не залежить від обчислювальної складності кодування.

Відповідно, обчислювальна складність алгоритмів кодування і стиснення визначає співвідношення практичного і теоретично можливого рівня стиснення.

Що є підставою запропонованої концепції адаптивних кодеків різного рівня складності: для кожного рівня обчислювальних ресурсів (обмежень) O є

оптимальний кодек, що при заданому коефіцієнті стиснення *K* і рівня ентропії *E* вхідних даних досягає оптимального практично можливого рівня стиснення.

Інші закономірності запропонованої математичної моделі стосуються процесу кодування і стиснення з втратами, а саме – залежності деформації вхідної інформації від коефіцієнта стиснення (тобто параметру квантування, що визначає коефіцієнт стиснення) і порядку перетворення.

Для цього представимо вхідні дані у формі суми функцій, які відповідають певним частотам, та згрупуємо їх в декілька груп, наприклад, високих, середніх і низьких частот:

$$g_i(x, y) = \alpha_i + \beta_i + \gamma_i,$$

α_i – група функцій відображення низьких частот;

β_i – група функцій відображення середніх частот;

γ_i – група функцій відображення високих частот.

Розглянемо процеси, які відбуваються при збільшенні порядку перетворення і при збільшенні параметра квантування, а також їх взаємодії.

Збільшення порядку перетворення означає збільшення локального масштабу відображення – якщо перетворення порядку 4 працює з блоком з 16 пікселів, то перетворення порядку 8 враховує 64 пікселі, а порядку 32 – 1024 пікселі. Відповідно, кількість високих частот у блоку 32x32 значно вища, ніж у блоці 4x4. В той же час низькі частоти краще враховує перетворення порядку 32.

З одного боку, це загальновідомі факти – перетворення великого порядку краще враховує низькі і середні частоти, а перетворення малого порядку – високі частоти, середнього порядку – середні частоти. З іншого боку, це не використовується для алгоритмів вибору перетворень, задовольняючись емпіричним перебором та підбором перетворення оптимального порядку (хоча це не тільки теоретичні, але й статистичні дані, які нескладно отримати, досліджувати й потім використати).

При цьому коефіцієнти високих частот швидко зменшуються зі збільшенням параметра квантування QP, чим більший коефіцієнт *C* стиснення даних, тим більше втрачається інформація про високі частоти:

$$\gamma_i(C) = k / C .$$

Чим більше розмір блоку і порядок перетворення *S*, тим більше втрачається інформації про високі частоти:

$$\gamma_i(S) = k / S$$

Тобто для блоків зображень, для яких переважають низькі частоти, краще використовувати перетворення великого порядка, де середні частоти – перетворення середнього порядка, де високі частоти – перетворення низького порядка. З урахуванням обох вищевказаних залежностей універсальнішими є перетворення середнього порядка для середніх коефіцієнтів стиснення.

Ця модель відображає особливості кодування вхідних даних без врахування попередньої обробки даних алгоритмами стиснення без втрат (тобто алгоритмами передбачення значень пікселів). Останні збільшують ваговий коефіцієнт *P* і зменшують ваговий коефіцієнт суми функцій $g_i(x, y)$, тобто збільшується ентропія оброблених даних (що відразу відображається у помітному зменшенні значень PSNR в разі застосування алгоритмів передбачення). Але додаткове збільшення загального коефіцієнту стиснення є підставою для використання алгоритмів передбачення, незважаючи на зменшення PSNR (аналогічно тому, як використання простору кольорів YCrCb зменшує об’єм даних з одночасним зменшенням PSNR).

Використання алгоритмів передбачення зменшує вплив розглянутих в моделі залежностей, але вони все одно виконуються і можуть бути використані для розробки нових підходів і алгоритмів.

В даній роботі результатом застосування розробленої математичної моделі є концепція адаптивних режимів різного рівня складності, алгоритм вибору порядку перетворення в залежності від статистичних характеристик вхідних даних і алгоритм вибору типу перетворення в залежності від параметру квантування коефіцієнтів перетворення.

2.3 Швидкодійний кодек стиснення зображень/відео

Кодек зображень має дві основні характеристики – коефіцієнт стиснення і обчислювальна складність. Додаткові параметри, наприклад, наявність прогресивної передачі, альфа-канал та інш. ми не беремо до уваги.

Коефіцієнт стиснення – співвідношення між оригінальним зображенням і відновленим при певному рівні похибки, *K*:1 раз. Також для оцінки коефіцієнту стиснення використовують параметр біт на піксель, тобто співвідношення об’єму даних до кількості пікселей растрового зображення.

Обчислювальна складність може бути оцінена в кількості операцій на піксель, окремо для кодування і декодування, або в тривалості обробки зображення/кадру на конкретному пристрої. Відповідно, є мінімальні вимоги кодека кожного покоління до обчислювальної потужності.

Визначення. Кодек відноситься до класу швидкодійних кодеків, якщо в пріоритеті зменшення обчислювальної складності, а не збільшення коефіцієнту стиснення.

Іншим варіантом є адаптивні режими з пріорітетом на швидкодії, що можуть бути складовими будь-яких кодеків. Таким чином, спеціалізовані кодеки і універсальні кодеки являють собою два можливих підхода до створення і використання кодека.

Переваги універсальних кодеків врівноважуються необхідністю додаткових алгоритмів вибору між різними адаптивними режимами. Відповідно, для умов з постійними обмеженнями на обчислювальні ресурси доцільно

використовувати спеціалізовані кодеки – швидкодійні або збалансовані, а для умов зі значними обчислювальними ресурсами – універсальні кодеки.

Важливим параметром є обернена сумісність кодеків, коли складніші кодеки здатні декодувати дані, закодовані швидкодійним кодеком.

Адаптивні режими, як і спеціалізовані кодеки, можуть мати наступні особливості реалізації:

- ✓ Пріорітет на швидкодію;
- ✓ Пріорітет на коефіцієнт стиснення (ресурсоємні);
- ✓ Збалансованість між швидкодією та стисненням.

Розглянемо деякі важливі підходи, методи і алгоритми для розробки швидкодіючих кодеків різного рівня.

2.3.1 Універсальний алгоритм передбачення значень пікселів

Алгоритми передбачення значень пікселів мають декілька різних реалізацій:

- ✓ для окремих пікселів;
- ✓ для блоків пікселів;
- ✓ для кадрів.

Відповідно, алгоритми для окремих пікселів були найпершими, наприклад, у формі різниць між сусідніми пікселями (так званий дельта-алгоритм), потім розробили алгоритми передбачення значень пікселів кадру на основі попереднього кадру (тобто обчислювали різниці між подібними блоками різних кадрів), і останнім винайшли передбачення значень пікселів поточного блоку на основі значень сусідніх блоків.

Хоча ідея обчислення різниць між блоками різних кадрів дуже подібна до ідеї обчислення різниць сусідніх блоків одного кадру/зображення, але остання була застосована значно пізніше, починаючи зі стандарту відеокодування H.264 в 2003 р. Відповідно, було запропоновано декілька різних алгоритмів передбачення значень пікселів (рис. 8), що обумовлює необхідність їх порівняння та вибору оптимального.

Крім того, спочатку обирається розмір блоку передбачення, а вже потім для конкретного розміру блоку – порівнюються алгоритми передбачення. Для того ж H.264 для базового профіля використовуються блоки двох типів – 4x4 і 16x16, а для high профіля – вже блоки трьох розмірів 4, 8 і 16, що додатково збільшує складність кодування. З метою зменшення обчислювальної складності для блоку розміром 16x16 застосовується лише 4 алгоритми передбачення значень пікселів. Але необхідність врахування і порівняння блоків передбачення розміром 16x16 збільшує загальну обчислювальну складність.

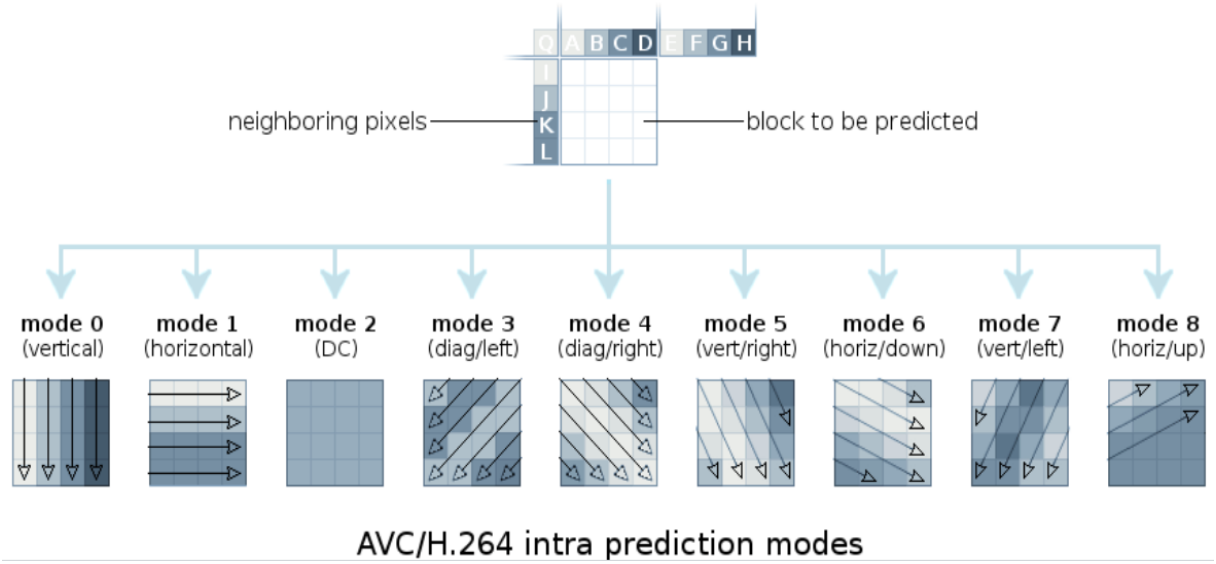

Рисунок 8 Алгоритми передбачення значень пікселів

З метою підвищення швидкодії (за рахунок незначного зменшення коефіцієнта стиснення) пропонується використовувати спрощені режими передбачення значень пікселів, тобто зменшену кількість алгоритмів передбачення та зменшену кількість розмірів блоків передбачення.

В граничному випадку це означає використання лише одного алгоритму передбачення значень пікселів, що дає підстави назвати його універсальним алгоритмом передбачення значень пікселів.

В роботі [7] було досліджено вплив застосування одного універсального алгоритму передбачення на коефіцієнт стиснення, з метою оцінки доцільності використання. Було показано, що застосування лише одного простого універсального алгоритму передбачення дає на 29–36.8% збільшення коефіцієнта стиснення для найкращого за якістю перетворення порядка 8 при

мінімальному збільшенні обчислювальних ресурсів (за рахунок обчислення додаткового оберненого блочного перетворення для сусіднього блока). Для тестування були взяті однонормові цілочислові косинусні перетворення порядку від 4 до 32, які застосовуються в стандарті відеокодування H.265.

Результати тестування за характеристикою PSNR для варіанту з передбаченням в просторі RGB наведені в табл.1. Як бачимо, найкраще для середньокорельованого зображення є перетворення порядку 8, на другому місці – порядку 4, на третьому – порядку 16. Перетворення порядку 32 має найгірші результати з усіх за якістю, і дуже близьке за коефіцієнтом стиснення до перетворення порядку 16, що означає недоцільність його адаптивного використання для швидкодійних кодеків (але для ресурсоємних кодеків воно використовується, оскільки є локально оптимальним для областей зображень з високою кореляцією).

Таблиця 1

Порядок 2D перетворень	Результати кодування за характеристикою PSNR (дБ)		
	QP 22	QP 27	QP 32
4	42.31	38.13	34.35
8	**42.80**	**38.78**	**34.46**
16	42.20	37.95	34.22
32	41.63	37.51	33.80

При тестуванні з передбаченням перетворення порядка 16 показало високі результати за коефіцієнтом стиснення порівняно з варіантом без передбачення – від 41 до 48 %, тобто коефіцієнт стиснення приблизно на 12 % більший, ніж для перетворення порядка 8, незважаючи на те, що передбачення для блока порядка 16 є менш точним (оскільки блок передбачення дорівнює блоку перетворення, відповідно, для блока розміру 4x4 точність передбачення буде найвищою, а для блоків порядка 16 та 32 – найменшою).

Враховуючи, що обчислення алгоритму передбачення для блока порядка 16 дорівнює обчисленню чотирьох алгоритмів передбачення порядка 8,

перетворення порядка 16 може бути рекомендовано для режиму більшого стиснення за меншої обчислювальної складності алгоритмів передбачення.

В цілому найбільший абсолютний коефіцієнт стиснення з використанням передбачення було отримано для перетворення порядку 16 (див. табл. 2).

Для блоків (і зображень) з високою кореляцією між пікселями перетворення порядка 16 буде мати ще вищий коефіцієнт стиснення.

Таблиця 2

Параметр квантування QP	Якість кодування за коефіцієнтом стиснення (K:1) для 2D перетворень порядка		
	4	8	16
22	3,73	4,24	4,36
27	4,89	5,26	5,82
32	6,56	7,25	8,08

Для тестування було вибрано тестове зображення «храм» з середньою кореляцією між пікселями, як найбільш типове (рис. 9). Перетворення порядку 4 має найнижчі показники за коефіцієнтом стиснення, тому його доцільно взагалі не використовувати.

Наведемо результати тестування для двох інших зображень, «фото» з високою кореляцією між пікселями (рис. 10) та «пейзаж» (рис. 11) з низькою кореляцією. Але зробимо це для режиму 4:4:4 в просторі кольорів YcbCr.

Рисунок 9 Тестове зображення «храм» 512x512 пікселів

Як відомо, використання простору кольорів YcbCr замість простору RGB зменшує якість за характеристикою PSNR приблизно на 1 дБ. за рахунок зменшення точності червоного і синього кольорів (замість рівномірного розподілу з ваговим коефіцієнтом 0,33 використовуються розподіли з акцентом на зелений колір і найменшим вкладом синього кольору).

Застосування YcbCr дає додаткове стиснення, тому на даний час вважається доцільним в більшості випадків (що може з часом змінитись – стандарти, вимоги якості постійно підвищуються).

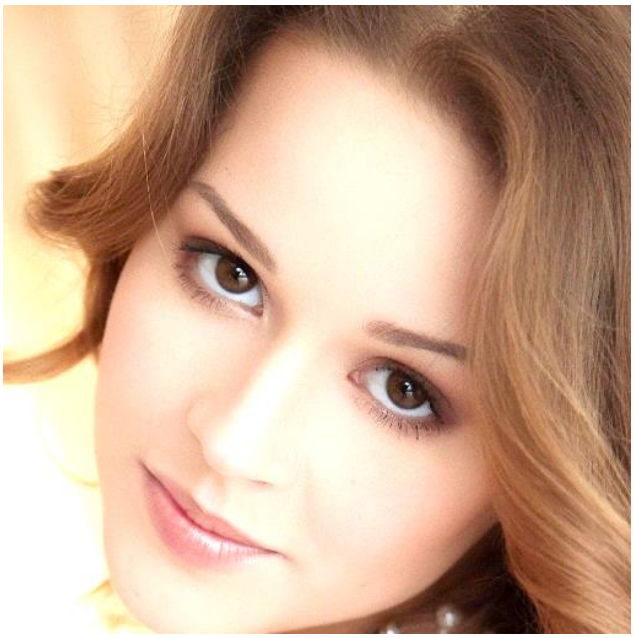

Рисунок 10 Тестове зображення «фото» 512x512 пікселів

Рисунок 11. Тестове зображення «пейзаж» роздільною здатністю 512x512.

Отже, порівняємо два перетворення порядку 8 та 16 (як найбільш ефективні для швидкодійного кодека) за параметрами PSNR та коефіцієнт стиснення. Для зображення «пейзаж» з використанням алгоритму передбачення результати тестування наведено в табл. 3., а без передбачення – в табл. 4.

Таблиця 3

Порядок 2D перетворень	Якість кодування за характеристикою PSNR (dB) для параметру квантування QPst		
	8	14	24
8	**39.41**	**35.29**	**30.84**
16	39.20	35.26	30.84

Таблиця 4

Порядок 2D перетворень	Якість кодування за характеристикою PSNR (dB) для параметру квантування QPst		
	8	14	24
8	**39.73**	**35.38**	**31.04**
16	39.40	35.37	30.90

Як бачимо, спостерігається різниця приблизно в 0,2-0,3 дБ., яка є найбільшою для QPst=8, тобто для найменшого коефіцієнта квантування/стиснення. Зауважимо, що QPst=8 відповідає QP=22, а QPst=14 QP=27, і QPst=24 QP=32. QPst дає більш точне уявлення про коефіцієнт квантування, ніж параметр QP, оскільки він і є реальний коефіцієнт квантування.

В табл. 5 і 6 наведено розміри реальних файлів коєфіцієнтів перетворення після квантування і заокруглення до цілих чисел та застосування алгоритмів ентропійного стиснення/кодування.

Таблиця 5

Порядок 2D перетворень	об’єм файлу без передбачення, кб. для параметру квантування QPst		
	8	14	24
8	251	205	162
16	226	183	144

Таблиця 6

Порядок 2D перетворень	об’єм файлу з передбаченням, кб. для параметру квантування QPst		
	8	14	24
8	221	174	131
16	209	167	127

Для зображення «фото» з високою кореляцією між пікселями аналогічні результати тестування наведено в табл. 7, 8, 9 і 10. Як бачимо, різниці за параметром PSNR для варіантів з використанням алгоритму передбачення і без для зображення «фото» є більшими, що пояснюється більш високою кореляцією між пікселями у порівнянні з зображенням «пейзаж».

Таблиця 7

Порядок 2D перетворень	Якість кодування за характеристикою PSNR (dB) для параметру квантування QPst		
	8	14	24
8	**40.78**	37.70	35.55
16	40.76	**37.89**	**35.82**

Таблиця 8

Порядок 2D перетворень	Якість кодування за характеристикою PSNR (dB) для параметру квантування QPst		
	8	14	24
8	**41.10**	**38.14**	**36.14**
16	41.02	38.06	36.08

Таблиця 9

Порядок 2D перетворень	об’єм файлу без передбачення, кб. для параметру квантування QPst		
	8	14	24
8	125	80.2	52.6
16	105	61.9	35.9

Таблиця 10

Порядок 2D перетворень	об’єм файлу з передбаченням, кб. для параметру квантування QPst		
	8	14	24
8	93.8	54.6	31.1
16	87.1	47.3	25.1

Також можна зробити висновок, що різниці між коефіцієнтами стиснення з використанням алгоритму передбачення і без нього зменшуються для зображень з низькою кореляцією, як і передбачується теорією.

Таким чином, використання навіть одного алгоритму передбачення значень пікселів дає суттєве збільшення коефіцієнту стиснення (яке залежить від кореляції початкових даних), при незначному збільшенні обчислювальної складності. Фактично більша частина підвищення ефективності кодека H.264 порівняно з H.263 досягається за рахунок використання алгоритмів передбачення значень пікселів, причому різниця між одним алгоритмом передбачення і адаптивним пошуком оптимального з декількох незначна.

Проміжним варіантом збільшення швидкодії є використання меншої кількості алгоритмів порівняно з іншими режимами кодування для блоку одного розміру і зменшення кількості розмірів блоків, що адаптивно обираються (та порядку перетворень). Найшвидшим варіантом буде застосування фіксованого перетворення порядку 8 або 16 до всього зображення, в залежності від статистичних характеристик зображення.

2.3.2 Спрощені перетворення зі швидкими алгоритмами

Іншим джерелом підвищення швидкодії є використання перетворень зі швидкими алгоритмами обчислень. В роботах [39, 40, 42] було запропоновано найбільш оптимальні швидкі алгоритми для цілочислових косинусних

перетворень порядку 8, 16 і 32. Зауважимо, що найбільш швидкодіючим за параметром кількість операцій на піксель є перетворення порядку 8.

З метою збільшення швидкодії доцільно застосовувати альтернативні косинусним перетворення, які мають меншу обчислювальну складність – слент-перетворення, перетворення Уолша, Уолша-Адамара, Хаара, Уолша-Хаара, а також так звані прості косинусні перетворення та ступінчасті косинусні. Всі ці перетворення будемо називати спрощеними, оскільки їх обчислювальна складність помітно менша, ніж в косинусних перетвореннях.

Збільшена швидкодія альтернативних спрощених перетворень супроводжується менш ефективною акумуляцією інформації в найбільших коефіцієнтах, тобто менш ефективним стисненням, в середньому на 5-10%. Але застосування алгоритмів перетворення зменшує різницю в коефіцієнті стиснення між косинусними перетвореннями і спрощеними.

Швидкі алгоритми обчислення перетворень використовуються не лише на етапі перетворення, але й на етапі передбачення значень пікселів, етапі вибору перетворення та на етапі оптимізації співвідношення бітрейта і похибки (RDO), тобто співвідношення коефіцієнта стиснення і похибки. Хоча для спеціалізованого кодека з пріорітетом на швидкодії ці етапи значно спрощуються, і залишається лише одне додаткове застосування – для обчислення відновлених значень сусідніх блоків пікселів, які необхідні для обчислення передбачення значень поточного блоку.

2.3.3 Алгоритм вибору типу перетворення

Чим більший коефіцієнт стиснення, тим менша різниця між ефективністю кодування косинусних та спрощених перетворень. Відповідно, для підвищення швидкодії є обгрунтованим використання спрощених перетворень, особливо для середніх та великих коефіцієнтів стиснення, тобто для QPst=24 і більше.

Іншими словами, пропонується новий алгоритм вибору типу перетворення в залежності від двох параметрів – коефіцієнта квантування і кореляції даних, а

саме: при великих значеннях коефіцієнта квантування (більше чи рівне 24) і при середній та високій кореляції початкових даних доцільно використовувати спрощені перетворення, наприклад, прості косинусні перетворення.

Для низької кореляції пікселів зображення/кадру при великих коефіцієнтах стиснення доцільно застосовувати косинусне перетворення порядку 16 (для середнього стиснення – косинусне перетворення порядку 8).

2.3.4 Спрощений вибір перетворень порядку 8 та 16

Як було сказано, найменшу обчислювальну складність має фіксований вибір порядку перетворення, тобто 8 або 16 в якості базового перетворення. Косинусне перетворення порядку 8 було взято за базове для кодеків JPEG, H.263 та інших (хоч і у формі ДКП, а не цілочисловій апроксимації зі швидкими алгоритмами, як у H.264). В кодеку H.265 згідно статистики використання перетворень найчастіше використовуються косинусні перетворення порядку 8 та 16, що також проілюстровано тестуваннями перетворень різного порядку від 4 до 32 вище, в підрозділі 2.3.1. Перетворення порядку 4 є неефективним за критерієм коефіцієнта стиснення, хоча має низьку похибку, а перетворення порядку 32 має занадто велику похибку для більшості зображень (областей зображень).

Відповідно, косинусне перетворення порядку 8 адаптивно обирається для областей зображень з низькою кореляцією між пікселями, а порядку 16 – для областей зображень з високою кореляцією між пікселями. Це особливо добре помітно у випадку застосування цих перетворень без алгоритмів передбачення значень пікселів, оскільки застосування алгоритмів передбачення збільшує ентропію, рівень хаотичності значень пікселів після передбачення (тобто різниць між поточними значеннями і прогнозованими значеннями). Іншими словами, ефективність застосування косинусних перетворень після алгоритмів передбачення значно зменшується, що ілюструється зменшенням значень PSNR при рівних значеннях коефіцієнта квантування. Але загальний коефіцієнт

стиснення в цілому збільшується, тому застосування алгоритмів передбачення (які є стисненням без втрат) є доцільним.

Незважаючи на зменшення ефективності косинусних перетворень у випадку застосуваннння алгоритмів передбачення, адаптивна ефективність перетворень різних порядків залишається помітною, тому є доцільним розробити та застосовувати простий алгоритм адаптивного вибору порядку перетворення в залежності від статистичних характеристик конкретного зображення/кадру.

Тобто замість адаптивного локального вибору порядку перетворення для кожного блоку, що потребує додаткових обчислень для порівняння та вибору, на основі кластерного аналізу можна розділити всі зображення на три групи:

1) зображення, де косинусне перетворення порядку 8 використовується частіше за косинусне перетворення порядку 16;
2) зображення, для яких косинусне перетворення порядку 16 використовується частіше за перетворення порядку 8;
3) зображення, де частота адаптивного вибору перетворень порядку 8 та 16 приблизно рівні.

Відповідно, потрібен критерій класифікації зображення/кадру до однієї з цих груп зображень, або декілька критеріїв різної складності, які можуть застосовуватись адаптивно, в залежності від обмежень на обчислювальну потужність.

Розглянемо найпростіший критерій з мінімальною обчислювальною складністю для оцінки кореляції між пікселями. В першому наближенні для цього можна скористатись дельта-алгоритмом обчислення різниць між сусідніми значеннями пікселів, сума яких дозволить оцінити кількість значних перепадів між пікселями, тобто контурів. Різниці обчислюються як по вертикалі, так і по горизонталі. Відношення суми всіх різниць до максимально можливої суми всіх значень пікселів зображення/кадру дозволить приблизно оцінити рівень кореляції між пікселями та віднести зображення до однієї з трьох груп:

$$K_k = \frac{\sum_{i,j=1}^{N} \left(\left| p_{i,j} - p_{i+1,j} \right| + \left| p_{i,j} - p_{i,j+1} \right| \right)}{2 * 256 * N^2},$$

де N – кількість пікселів по вертикалі і горизонталі (для зображення з роздільною здатністю $N * N$).

Для оцінки діапазону значень параметру K_k для кожної з груп потрібно зібрати та проаналізувати статистичні дані тестування масиву зображень, що планується зробити в наступних дослідженнях.

Для оцінки рівня кореляції і вибору порядку перетворення для всього зображення обчислювальна складність зростає мало, на 1-1,2 операції на піксель, що у порівнянні з загальною складністю кодека становить біля 1%.

Ефективність застосування алгоритму вибору порядку перетворення для всього зображення у порівнянні з фіксованим значенням порядку перетворення можна оцінити на підставі збільшення ефективності стиснення перетворення порядку 16 відносно порядку 8, яке складає близько 12% (від 10 до 15) для середньої кореляції (ефективність застосування перетворення порядку 16 збільшується для високої кореляції вхідних даних і зменшується – для низької).

Також можна запропонувати альтернативний алгоритм: обчислювати замість всіх різниць тільки ті, які більше обраного значення X, і оцінювати кількість таких різниць до загальної кількості пікселів.

Більш складним і тому більш ефективним є адаптивний локальний вибір порядку перетворення за спрощеним алгоритмом. Але навіть для спрощеного алгоритму вибору порядку перетворення потрібно зробити певні попередні обчислення: зробити передбачення для блоку на 8; отримати оцінку кореляції, обчисливши K_k для конкретного блоку пікселів. Якщо отриманий параметр K_k буде вище певного порогового значення X (емпірично обчисленого на великій виборці зображень), то обирається перетворення порядку 8 (низька кореляція), а коли нижче – то перетворення порядку 16 (висока кореляція між пікселями).

Найбільшу ефективність адаптивний алгоритм вибору порядку

перетворення буде мати для зображень третьої групи, де кількість використань перетворень порядку 8 і 16 приблизно однакова. Отже, спочатку оцінюється параметр K_k для зображення в цілому, у випадку низької кореляції обирають лише перетворення порядку 8, у випадку високої кореляції – перетворення порядку 16, і у випадку середньої кореляції – адаптивний вибір перетворень.

2.3.5 Спрощені простори кольорів та вибір оптимального простору кольорів

Пріорітет на мінімізації обчислювальної складності кодека означає менший коефіцієнт стиснення вхідних даних для кожного рівня похибок, ніж у випадку збалансованого або ресурсоємного кодека. При наявності каналу зв'язку з достатньою пропускною здатністю це прийнятно, але режими кодування, в яких є додаткові вимоги по стисненню необхідно розглянути, врахувавши всі можливості збільшення коефіцієнту стиснення.

На додаток до збільшення параметра квантування отриманих коефіцієнтів перетворення, основного шляху збільшення коефіцієнта стиснення, який автоматично збільшує похибку відновлених даних, розглянемо інші підходи, методи і алгоритми.

Найвідомішим підходом є зменшення точності кольорів, наприклад, розповсюджене використання простору кольорів YCrCb та його режимів замість RGB дає збільшення коефіцієнту стиснення за рахунок деякого зменшення точності передачі кольорів, як в цілому (за рахунок зменшення вагової складової червоного та особливо синього кольорів) так і локально, в результаті застосування спрощених режимів YCrCb (4:2:2 і 4:2:0).

Стандарт кодування зображень GIF, як відомо, використовував так звану індексовану палітру кольорів, що дозволяло значно зменшити об'єм даних.

Іншим відомим підходом було використання спрощених просторів кольорів, коли на кожен канал кольору використовували не 8 біт, а меншу кількість – 4 або 5 чи 6. Відповідно, були 12 бітні зображення, або 16 бітні (6 біт на зелений колір, і по 5 – на червоний і синій, наприклад).

Але стандарт кодування по 8 біт на канал кольору швидко витіснив спрощені простори кольорів, включаючи індексовані палітри. Хоча задача оптимального розподілу біт залишається актуальною за умови значного стиснення – похибка для 24 бітних зображень може бути більшою, ніж для 16 бітних, при однаковому коефіцієнті стиснення (але при різних коефіцієнтах квантування – для 24 бітних зображень коефіцієнт квантування має бути більшим, ніж для 16 бітних, щоб отримати приблизно той же коефіцієнт стиснення).

Щодо індексованих палітр, то їх застосування є доцільним для інфографіки, тобто зображень з малою кількістю кольорів. Індексована палітра адаптивно використовувалась в більшості маловідомих кодеках зображень, а також в форматі PNG.

Але в останньому випадку індексована палітра мало використовувалась, оскільки PNG за замовчуванням використовував 24 бітний простір кольорів, а більшість користувачів не міняла налаштування для інфографіки. Алгоритм адаптивного вибору між індексованою палітрою і повним простором кольорів є достатньо простим, і може бути реалізовано автоматично, програмно.

Таким чином, адаптивне використання спрощених просторів кольорів та індексованих палітр є доцільним і обгрунтованим за умов значного обмеження обчислювальної потужності та пропускної здатності каналу.

З розвитком методів кодування і стиснення зображень з використанням нейромережевих технологій (ШІ) може бути доцільним використання спрощених просторів кольорів та значних коефіцієнтів стиснення з метою подальшої обробки та покращення якості зображень/кадрів з застосуванням ШІ.

Це найбільш ймовірний і обчислювально простий напрямок застосування ШІ, оскільки використання ШІ для збільшення коефіцієнту стиснення (тобто розробки нових підходів, методів і алгоритмів) потребує значно більших обчислювальних потужностей.

Хоча оптимізація та адаптивне використання існуючих алгоритмів стиснення може мати приблизно таку ж обчислювальну складність, як і покращення візуальної якості.

Зокрема, робота з кольором може стати одним з напрямків оптимізації алгоритмів стиснення, включаючи розробку нових просторів кольорів або гібридів просторів кольорів і індексованих палітр.

Іншим напрямком застосування спрощених просторів кольорів та індексованих палітр є так звані екстремальні режими кодування, які формуються на всяк випадок, для застосування при екстремально низьких параметрах передачі та стиснення. Розглянемо детальніше так звані екстремальні адаптивні режими стиснення, що зазвичай характеризуються великими коефіцієнтами стиснення при мінімальних обчислювальних витратах.

2.3.6 Екстремальні режими стиснення зображень з втратами

Розглянемо деякі підходи, методи і алгоритми для екстремальних режимів стиснення зображень та відеокадрів при дуже низьких значеннях пропускної здатності каналу та обчислювальних потужностях.

Для таких режимів характерно значні втрати початкової інформації при кодуванні і стисненні, що обумовлює задачу оптимізації використання ресурсів.

Одним із шляхів її вирішення є динамічна зміна параметрів кодування, зокрема, роздільної здатності, глибини кольору та інш.

Другим відомим напрямком є використання прогресивної передачі даних, тобто передача спочатку значно зменшеного зображення, потім різниць декількох рівнів, для отримання збільшенного, тобто початкового зображення/кадру.

Цей алгоритм є базовим для кодека LCEVC, отримавши його в рамках ідеї використання сторонніх кодеків для кодування зменшеного зображення, і використання власних алгоритмів – для кодування і стиснення різниць між зображеннями різного розміру.

Кодування окремо різниць і окремо зменшеного зображення принципово відрізняється від попереднього алгоритму прогресивної передачі, в якому коефіцієнти перетворення розділялись на дві і більше груп, частин і при оберненому перетворенні утворювали при інтеграції, об’єднанні початкове зображення.

Отже, базовим підходом до екстремальних режимів стиснення даних з втратами є значне обмеження інформації, тобто великі коефіцієнти стиснення і спрощені простори кольорів, включаючи відтінки сірого та чорно-білі зображення.

Застосування динамічної зміни параметрів одночасно з новим алгоритмом прогресивної передачі даних дозволить розширити діапазон переданої інформації, в тому числі за рахунок узгодження та аналізу даних.

2.4 Збалансований кодек стиснення зображень/відео

На відміну від класу швидкодійних кодеків, що мають пріорітетом швидкодію, клас збалансованих кодеків зображень/відео відрізняється пошуком балансу між обчислювальною складністю та коефіцієнтом стиснення. Оскільки поняття балансу з часом може значно мінятись, з урахуванням змін обчислювальних потужностей мікропроцесорів, критеріїв якості, пропускної здатності каналів зв’язку та інш., то можна говорити про еволюцію збалансованих кодеків різних поколінь.

2.4.1 Адаптивний вибір оптимальних перетворень

Збалансований кодек широко використовує локальну адаптивність алгоритмів, яка збільшує обчислювальну складність порівняно з фіксованими значеннями параметрів, але й збільшує коефіцієнт стиснення даних. Відповідно, є три етапи, де використовуються адаптивні алгоритми:

1. етап стиснення без втрат алгоритмами передбачення значень пікселів поточного блоку на основі значень сусідніх блоків зображення;
2. етап перетворень, на якому обирається тип і порядок перетворення;
3. етап квантування, на якому обирається коефіцієнт квантування.

Відповідно, на першому етапі обирається розмір блоку і варіант алгоритму передбачення, що дає мінімальні значення різниць між поточними значеннями пікселів і очікуваними, тобто передбаченими на основі сусідніх блоків.

Розмір блоку передбачення впливає на порядок перетворення, оскільки для передбачення використовуються не початкові значення пікселів сусідніх блоків, а відновлені, тобто з похибкою, і тому розмір матриці перетворення має бути не більшим за розмір блоку передбачення.

Для отримання відновлених значень сусідніх блоків потрібно зробити пряме і обернене перетворення, причому для оберненого перетворення потрібно мати конкретне значення параметру квантування коефіцієнтів перетворення, який підбирається в залежності від вимог до бітрейту (у випадку відео) або бажаного коефіцієнта стиснення, біт на піксель, для певного рівня прийнятної похибки.

Іншими словами, всі три етапи є пов’язаними, і потребують перебору та порівняння значної кількості можливих комбінацій пов’язаних значень параметрів алгоритмів. Оптимізація, підвищення ефективності процесу перебору та підбору оптимальних параметрів є складною задачею.

Одним з напрямків зменшення обчислювальної складності є використання прогресивної передачі даних, оскільки і для зменшених зображень, і для різниць між зображеннями різного масштабу статистика оптимальних перетворень значно змінюється: якщо для зображень з великою роздільною здатністю доцільним є використання косинусних перетворень порядку 32 (як в H.265) чи навіть 64 (як в H.266), то для зменшених зображень доцільність, як і частота їх використання, швидко зменшуються зі зменшенням розмірів зображення.

В свою чергу для різниць між зображеннями різного масштабу в кодеку LCEVC також використовуються перетворення малого порядку (додатково

використовуючи алгоритми фільтрації коефіцієнтів перетворення з урахуванням моделі перцепції людини).

Таким чином, використання прогресивної передачі даних додатково підвищує важливість й частоту вибору перетворень порядку 8 і 16, а також обумовлює використання універсального алгоритму передбачення значень пікселів на основі зовнішніх даних, тобто – даних зменшеного зображення, що разом складають основу для базового й найбільш оптимального адаптивного режиму збалансованого кодеку.

Але інші адаптивні режими (профілі) також необхідні і адаптивно використовуються за певних умов, а саме – швидкодіючі, спрощені режими низької обчислювальної складності і адаптивні режими високої обчислювальної складності. Розглянемо їх детальніше.

2.4.2 Адаптивні режими середньої і високої обчислювальної складності

Адаптивний режим середньої обчислювальної складності збалансованого кодека має використовувати найкращі підходи і алгоритми кодеків H.264 і H.265.

В кодеку H.264 для high профіля розроблено алгоритми адаптивного вибору цілочислових косинусних перетворень порядку 4 і 8 [6], а в кодеку наступного покоління H.265 використовується адаптивний вибір між чотирма косинусними перетвореннями порядків від 4 до 32 [26]. Жоден з цих варіантів не є оптимальним з точки зору оптимального співвідношення, балансу обчислювальної складності кодування і коефіцієнта стиснення.

Відповідно, для базового режиму збалансованого кодека (адаптивного режиму середньої обчислювальної складності) будемо вважати оптимальним вибір з двох цілочислових косинусних перетворень – порядку 16 і 8, що згідно статистичних даних найчастіше обираються в кодеку H.265 серед чотирьох косинусних перетворень. Перетворення порядку 4 має високу точність, внаслідок малого масштабу, але недостатній коефіцієнт стиснення.

Перетворення порядку 32 має високий коефіцієнт стиснення, але недостатньо високу точність на більшості зображень.

Адаптивний режим високої складності збалансованого кодека включає в себе використання трьох спеціалізованих режимів: для низької кореляції, для високої і універсальний режим. Спеціалізований режим для низької кореляції застосовує косинусне перетворення порядку 8. Для високої кореляції доцільно адаптивно застосовувати набір з двох перетворень, порядку 16 і 32, і універсальний режим адаптивно використовує три перетворення: 8, 16 і 32.

При цьому використання роздільних адаптивних перетворень, тобто комбінацій косинусного і синусного (або інших) перетворень не є доцільним, оскільки ефективність стиснення зростає близько 1% при значній додатковій обчислювальній складності. Роздільні адаптивні перетворення є доцільними лише для ресурсоємних кодеків, і будуть розглянуті нижче.

Для зменшення обчислювальної складності пропонується нова архітектура кодування, що використовує додатковий структурний, ієрархічний рівень, на якому порівнюються та обираються ті чи інші адаптивні режими. В результаті замість перебору та порівняння всіх можливих перетворень різних порядків обирається окремий адаптивний режим, в рамках якого йде порівняння й адаптивний вибір відповідних алгоритмів. Алгоритми вибору адаптивних режимів незначно збільшують складність кодування, але більшість адаптивних режимів мають значно меншу складність, що дозволяє оптимізувати загальну обчислювальну складність.

2.4.3 Алгоритми вибору адаптивного режиму і нова архітектура кодеків

Запропонована нова архітектура кодеків грунтується на загальній статистиці зображень, статистичному розподілі на декілька груп з різними характеристиками. Відповідно, оцінивши на початку кодування загальні статистичні характеристики зображення/кадру, можна звузити кількість перетворень для цього конкретного зображення. Ефективність цього підходу

залежить від однорідності статистичних характеристик окремих областей зображення. Іншими словами, узагальнений спрощений підхід оцінки зображення краще працює для групи зображень з високим рівнем однорідності, і гірше – для групи зображень з низьким рівнем однорідності областей. Для останьої групи застосовується універсальний режим. Альтернативним підходом є розбиття складного неоднорідного зображення/кадру на окремі області зі схожими статистичними характеристиками. В цьому випадку формується додаткова службова інформація про розподіл областей зображення, і до кожної області застосовується свій адаптивний режим, набір перетворень та алгоритмів. Алгоритм вибору адаптивного режиму базується на відомій закономірності – для кодування високочастотної інформації зі значними перепадами значень між сусідніми пікселями краще підходять перетворення меншого порядку, 4 та 8, а для кодування низькочастотної інформації з незначними перепадами значень між сусідніми пікселями краще підходять перетворення більшого порядку, 16 та 32. При цьому перетворення порядків 8 і 16 є більш універсальними, а 4 і 32 – більш спеціалізованими. Пропорція між кількістю значних перепадів і незначних визначає статистичні характеристики області зображення та самого зображення. Узагальнений алгоритм обчислення співвідношення застосовується для всього зображення, при цьому обчислюються різниці між сусідніми пікселями по вертикалі і по горизонталі (тобто часткові похідні першого рівня), а локальний алгоритм застосовується для певного блоку зображення, і може використовувати як різниці між пікселями, так і різниці між передбачуваними значеннями блоку і поточними значеннями.

2.4.4 Адаптивні режими низької складності

Збалансований кодек відрізняється від спеціалізованого швидкодійного кодека, але може включати останній в себе як складову частину.

Це збільшить діапазон адаптивних можливостей кодека, так само як екстремальні режими наднизької обчислювальної складності розширюють умови використання швидкодійких кодеків.

Крім того, якщо є можливість зробити обернену, зворотню сумісність двох кодеків, наприклад, при паралельній їх розробці, це виглядає доцільним і обгрунтованим. Таким чином, збалансований кодек може включати в себе і адаптивні режими низької складності, та вміти декодувати дані, закодовані швидкодійним кодеком.

Відповідно, виникає додаткова задача вибору адаптивних режимів. Пріорітетом і базовим режимом для збалансованого кодеку є адаптивні режими середньої та високої складності, тому адаптивні режими низької та наднизької складності включаються за певних умов неочікуваних обмежень обчислювальної складності чи пропускної здатності каналу зв’язку.

Задача порівняння і вибору оптимального адаптивного режиму ускладнюється для кодеків інших типів – ресурсоємного і універсального, особливо для останього, який включає в себе всі інші типи кодеків – швидкодійний, збалансований, ресурсоємний.

Розглянемо більш детально особливості ресурсоємного кодеку.

2.5 Ресурсоємний кодек стиснення зображень/відео

Більшість кодеків різних поколінь розроблялись з метою найбільшого стиснення вхідних даних, з використанням всіх доступних обчислювальних потужностей, або навіть з урахуванням подальшого збільшення. Це обумовлює їх неоптимальність з позицій сучасних уявлень і моделей.

Ресурсоємні кодеки використовують ряд підходів, методів і алгоритмів, що є недоцільними для швидкодійних та збалансованих кодеків.

Розглянемо їх детальніше, та запропонуємо деякі нові алгоритми для їх покращення.

2.5.1 Локальні адаптивні алгоритми

Локальні адаптивні алгоритми прийшли на зміну фіксованим алгоритмам, підбираючи найбільш оптимальний варіант параметрів кодування та алгоритмів під конкретні особливості блоків зображень.

Всі адаптивні алгоритми діляться на дві групи:

- з явними закономірностями між статистичними характеристиками вхідних даних та оптимальними алгоритмами;
- з неявними та невідомими закономірностями між статистичними характеристиками вхідних даних та вибором алгоритмів.

Перша група має певні виражені умови застосування під конкретні особливості блоків зображень, наприклад – рівень кореляції пікселів для вибору порядку перетворень, а друга група потребує безпосереднього тестування і порівняння для вибору найкращого.

Більшість адаптивних алгоритмів належать саме до другої групи, що і обумовлює значну обчислювальну складність для досягнення максимально можливого коефіцієнту стиснення ресурсоємних кодеків при заданому рівні похибки. Загальна тенденція на збільшення візуальної якості та точності зображень та відео збільшує об’єми закодованих та переданих даних, незважаючи на збільшення ефективності стиснення.

Основним напрямком збільшення ефективності стиснення є збільшення кількості адаптивних алгоритмів, що автоматично збільшує й обчислювальну складність порівняння та вибору одного з них. Наприклад, якщо у кодеку H.264 використовували 9 алгоритмів передбачення значень пікселів поточного блоку на основі сусідніх блоків, то в H.265 – вже 35 алгоритмів, а в H.266 – 67 алгоритмів. Крім того, збільшується кількість розмірів блоків для передбачення, а також кількість перетворень. Кодек наступного покоління, H.267, планується розробити з використанням нейромереж, що також значно збільшить обчислювальну складність. З іншого боку, застосування нейромереж може

оптимізувати й адаптивні режими середньої та низької складності. Розглянемо питання адаптивних режимів та вибір оптимального для ресурсоємних кодеків.

2.5.2 Адаптивні режими та вибір оптимального режиму

На відміну від збалансованого кодека, де вибір між різними адаптивними режимами є доцільним і дозволяє підібрати спеціалізований адаптивний режим меншої обчислювальної складності, для ресурсоємного кодека додатковий структурний рівень виглядає зайвим, оскільки лише емпіричний перебір та порівняння всіх можливих комбінацій дає можливість отримати найбільший коефіцієнт стиснення для заданого рівня похибки.

Але на практиці ресурсоємні кодеки змушені враховувати існуючі обмеження, особливо для процесу кодування, коли конкретні параметри обчислювальної потужності кодуючих пристроїв та пропускної здатності каналу зв’язку накладають обмеження на час кодування в режимі реальної обробки і передачі даних (особливо для режимів зі збільшеною кількістю кадрів в секунду). Отже, для ресурсоємних кодеків також є доцільним і обгрунтованим застосування додаткового рівня вибору, на якому відбувається вибір наборів алгоритмів, з подальшим адаптивним порівнянням і вибором алгоритмів вже в рамках конкретного набору алгоритмів.

Пріорітет на збільшенні коефіцієнта стиснення накладає свої особливості як на алгоритми вибору адаптивних режимів, так і на самі адаптивні режими. Ці питання потребують подальших досліджень, особливо в напрямку підвищення ефективності, оскільки ідея адаптивних комплексів алгоритмів розроблялась саме для зменшення обчислювальної складності кодування.

Відповідно, саме на зменшенні обчислювальної складності ресурсоємних кодеків ми й зупинимось в рамках даної роботи. Для цього ми згадаємо деякі нові алгоритми, що було розглянуто вище. Наприклад, застосування алгоритмів прогресивної передачі даних. Які пропонують алгоритми передбачення значень пікселів не на основі сусідніх блоків, а на основі значень пікселів зменшених

зображень/кадрів, тобто зовнішніх даних, зі значно меншою обчислювальною складністю. Крім того, концепція прогресивної передачі даних для зображень передбачає новий адаптивний режим передачі за вимогою. В цьому випадку потрібно зберігати додаткову інформацію про оригінальне зображення і мати доступ до каналів зв'язку, аби за необхідності надіслати додатковий рівень деталізації зображення.

Інший підхід для зменшення обчислювальної складності адаптивних режимів ресурсоємного кодування – відмова від неефективних перетворень порядку 4, які мають менший коефіцієнт стиснення ніж перетворення порядку 8. Для ресурсоємних кодеків пропонується три альтернативні набори перетворень (адаптивні режими) – для низької кореляції 8 і 16, для високої кореляції 16 і 32, для середньої кореляції універсальний адаптивний режим 8, 16, 32.

З урахуванням третього підходу – зменшення кількості службової інформації, доцільно відмовитись також від роздільних адаптивних (направлених) перетворень порядку 4, які вимагають зберігання надто великих об'ємів службових даних, і використовувати роздільні адаптивні перетворення починаючи з порядку 8. Для збільшення коефіцієнта стиснення в ресурсоємних кодеках пропонується новий локальний адаптивний алгоритм роздільних адаптивних перетворень з використанням нового типу перетворень – синус-косинусних. Розглянемо це питання детальніше.

2.5.3 Локальний адаптивний вибір типу перетворення в роздільних адаптивних перетвореннях

Новий запропонований алгоритм є продовженням концепції пошуку найбільш ефективного локального представлення даних різними типами перетворень. Якщо в цілому статистично косинусні перетворення мають краще співвідношення коефіцієнт стиснення/похибка за синусні перетворення, то локально існують такі особливості даних, що краще стискаються синусними, ніж

косинусними, або їх комбінаціями, що було названо роздільними адаптивними (направленими) перетвореннями.

Подальший розвиток цього підходу привів до дослідження і використання комбінацій перетворень більших порядків, що збільшило обчислювальну складність, оскільки обумовило емпірично оцінювати та порівнювати ще більшу кількість варіантів.

Пропонується новий алгоритм, що використовує не тільки косинусні і синусні перетворення для пошуку найкращої комбінації, але й синус-косинусні перетворення. Відповідно, це приводить до збільшення кількості варіантів перебору, і, звісно, означає збільшення обчислювальної складності.

Також пропонується замість синусного перетворення використовувати синус-косинусне перетворення в схемі роздільних адаптивних перетворень.

На жаль, поки що не існує алгоритму оцінки статистичних характеристик зображення, щоб можна було передбачити перевагу синусного або синус-косинусного перетворення для конкретного зображення/кадру. Відповідно, адаптивні режими поповнюються лише одним додатковим адаптивним режимом підвищеної складності – з використанням трьох перетворень замість двох.

При цьому використовується додатковий рівень вибору адаптивних режимів – тобто потрібен алгоритм вибору між двома режимами, використанням двох і трьох перетворень. Який залежить від оцінок обчислювальних ресурсів.

2.6 Універсальний кодек стиснення зображень/відео

Розглянемо наступний клас кодеків, та його особливості.

Означення. Кодек, що включає в себе всі три типи спеціалізованих кодеків (швидкодійні, збалансовані, ресурсоємні) як складові частини, будемо називати універсальним кодеком.

Універсальний кодек має свої недоліки – підвищену складність вибору адаптивних режимів і збільшення об’єму програмного коду. Відповідно, застосування універсального кодеку замість спеціалізованого швидкодійного є

недоцільним, особливо за умов постійних обмежень на обчислювальну потужність.

Універсальний кодек є альтернативою ресурсоємних кодеків при умовах можливості, ймовірності значних коливань обчислювальної потужності чи її обмежень.

2.6.1 Алгоритми вибору адаптивних режимів

Алгоритми вибору адаптивних режимів для універсального кодека мають підвищену складність порівняно з іншими типами кодеків. Універсальні кодеки найкраще ілюструють нову запропоновану архітектуру кодування з вибором наборів алгоритмів і подальшим адаптивним вибором алгоритмів. Можна розглядати більш ускладнений варіант, що включає рівень вибору профіля кодеку. Тобто спочатку обирається швидкодійний, збалансований або ресурсоємний профіль кодування, і вже в рамках обраного профіля обирається той чи інший набір алгоритмів, адаптивний режим.

Універсальний кодек може мати додаткові налаштування і пріорітети, відповідно, при одних і тих же зовнішніх параметрах і умовах можуть прийматись різні рішення щодо вибору адаптивних режимів.Тобто універсальний кодек може бути ближчим до будь-якого зі спеціалізованих кодеків. Також зі зростанням вимог до якості і точності відновлення даних алгоритми вибору адаптивних режимів можуть модифікуватись.

Загальні принципи вибору адаптивних режимів залишаються ті ж самі, що й для швидкодійних, збалансованих і ресурсоємних кодеків.

2.6.2 Зворотня сумісність кодеків різного рівня

Спеціалізовані кодеки різного рівня складності можуть мати зворотню сумісність за умови їх паралельної розробки та узгодження адаптивних режимів. Враховуючи значне збільшення складності з кожним поколінням, включення

попередніх кодеків в якості складових частин не буде займати багато ресурсів. Зворотня сумісність між двома близькими кодеками є більш обгрунтованою: використання швидкодійних режимів для збалансованого кодека, або використання адаптивних режимів збалансованого кодека як частини ресурсоємного кодека.

Окремо стоїть випадок універсального кодека, який по замовчуванню включає в себе всі спеціалізовані кодеки, тобто має не тільки зворотню сумісність з ними, але й частково пряму (тобто закодовані універсальним кодеком дані можуть декодувати спеціалізовані кодеки).

2.7 Висновки до розділу

В розділі описано нову математичну модель кодування і стиснення зображень/кадрів з втратами. На основі запропонованої матмоделі зроблено висновок про існування оптимальних кодеків для кожного рівня обчислювальної потужності, який став підставою для формулювання нової концепції адаптивних кодеків різного рівня складності – швидкодійних, збалансованих, ресурсоємних і універсальних.

В рамках нової концепції адаптивних кодеків різного рівня запропоновано включення адаптивних режимів низької складності, що використовуються для швидкодійного кодека, в кодеки вищих рівнів в якості складової частини. Необхідність вибору і переключення між адаптивними режимами різної складності є підставою для введення додаткового ієрархічного рівня в структурі кодека, що породжує нову архітектуру кодування. Для швидкодійного кодека в якості альтернативі режимам низької складності додано так званий екстремальний режим низької складності, який має ще менші мінімальні вимоги до обчислювальних потужностей.

Використання в кодеках вищих рівнів складності адаптивних режимів, що використовуються в кодеках попередніх рівнів складності, обгрунтовує можливість зворотньої сумісності кодеків.

В рамках розвитку концепції адаптивних кодеків різного рівня для кожного класу кодеків запропоновано ряд нових алгоритмів.

Для швидкодійного кодека запропоновано наступні алгоритми, модифікації та покращення у порівнянні з кодеком H.263:

- універсальний алгоритм передбачення значень пікселів;
- цілочислові косинусні перетворення зі швидкими алгоритмами;
- прості цілочислові косинусні перетворення порядку 16 зі швидкими алгоритмами без операцій множення для окремих режимів кодування;
- алгоритм вибору перетворень в залежності від коефіцієнта стиснення;
- простий алгоритм адаптивного вибору перетворень порядку 8 і 16 для всього зображення/кадру;
- екстремальні режими кодування наднизької обчислювальної складності;
- використання спрощених просторів кольорів.

Для збалансованого кодека запропоновано наступні алгоритми, модифікації і покращення відносно кодеків H.264 і H.265:

- адаптивні режими середньої і високої обчислювальної складності;
- алгоритми вибору адаптивних режимів;
- адаптивні режими низької обчислювальної складності;
- зворотня сумісність з швидкодійним кодеком;

Для ресурсоємного кодека запропоновано наступні алгоритми, модифікації і покращення відносно кодеків H.265 і H.266:

- алгоритм адаптивного використання трьох типів перетворень: косинусного, синусного і синус-косинусного;
- адаптивні режими високої складності;
- цілочислове модифіковане синус-косинусне перетворення порядку 8 зі швидкими алгоритмами низької обчислювальної складності;
- використання адаптивних режимів низької і середньої складності кодеків попереднього рівня;
- зворотня сумісність зі збалансованим кодеком.

Запропоновано об’єднання всі трьох типів кодеків в універсальний кодек. Універсальний кодек характеризується більшою складністю алгоритмів вибору адаптивних режимів, у порівнянні зі спеціалізованими кодеками.

На основі розробленої мат. моделі запропоновано нові алгоритми вибору типу й порядку перетворення в залежності від статистичних характеристик вхідних даних і від коефіцієнту стиснення (параметру квантування).

В розділі було розглянуто питання побудови спеціалізованих кодеків різної направленості нової архітектури, яка включає в себе додатковий рівень порівняння й вибору – рівень адаптивних режимів, тобто наборів взаємозв’язаних алгоритмів. Розглянуто питання мінімізації вимог до обчислювальних потужностей шляхом використання додаткових адаптивних режимів низької складності, включаючи так звані екстремальні режими. Запропоновано зворотню сумісність кодеків різного рівня, з метою можливості декодування більш складним кодеком закодованих даних кодеком попереднього рівня складності.

Обгрунтовано використання прогресивної передачі даних з універсальним алгоритмом передбачення на основі зовнішніх даних (зменшеного зображення).

Запропоновано простий алгоритм адаптивного вибору порядку перетворення в залежності від статистичних характеристик конкретного зображення/кадру, що дає збільшення ефективності стиснення на 10-12% у порівнянні з фіксованим вибором перетворення порядку 8. Для зображень зі змінною кореляцією запропоновано адаптивний локальний вибір порядку перетворення за спрощеним алгоритмом.

Запропоновано підхід використання спрощених просторів кольорів та значних коефіцієнтів стиснення з метою подальшої обробки та покращення якості зображень/кадрів з застосуванням ШІ.

РОЗДІЛ 3
ЦІЛОЧИСЛОВІ ПЕРЕТВОРЕННЯ ДЛЯ АДАПТИВНОГО ЗАСТОСУВАННЯ

Розглянемо та порівняємо з відомими запропоновані ортогональні блочні цілочислові перетворення зі швидкими алгоритмами обчислення для адаптивного вибору з урахуванням різних критеріїв.

3.1 Цілочислове просте косинусне перетворення порядку 16

В якості альтернативи косинусним перетворенням можна розглянути декілька різних класів перетворень, які можна об’єднати під спільною назвою спрощені перетворення, оскільки їх обчислювальна складність помітно менша за обчислювальну складність косинусних перетворень.

Зокрема, до спрощених перетворень відносяться прості косинусні перетворення, які мають серед базисних функцій частину косинусних функцій. Відповідно, прості косинусні перетворення мають деяку перевагу перед іншими альтернативними перетвореннями для апроксимації незначних перепадів, коливань значень пікселів. За обчислювальною складністю вони знаходяться посередині між косинусними та спрощеними перетвореннями типу Уолша-Хаара, слент, ступінчастих та інш.

3.1.1 Метод побудови цілочислового простого косинусного перетворення порядку16

В роботі [39] запропоновано методи побудови простих цілочисельних косинусних перетворень (ЦКП) двох типів порядку 16 на основі ступінчастих функцій. За допомогою зазначеного методу побудуємо просте ЦКП порядку 16 низької складності, в якого перші чотири непарні функції представляють собою 8-ми ступінчасті прості косинусні базисні функції, а решта непарних функцій є

4-х ступінчастими функціями цілочисельного ступінчастого перетворення (ЦСП). Парні функції представляють собою 4-х і 8-ми ступінчасті функції ЦСП. На рис. 12 представлені 16 базисних функцій простого ЦКП.

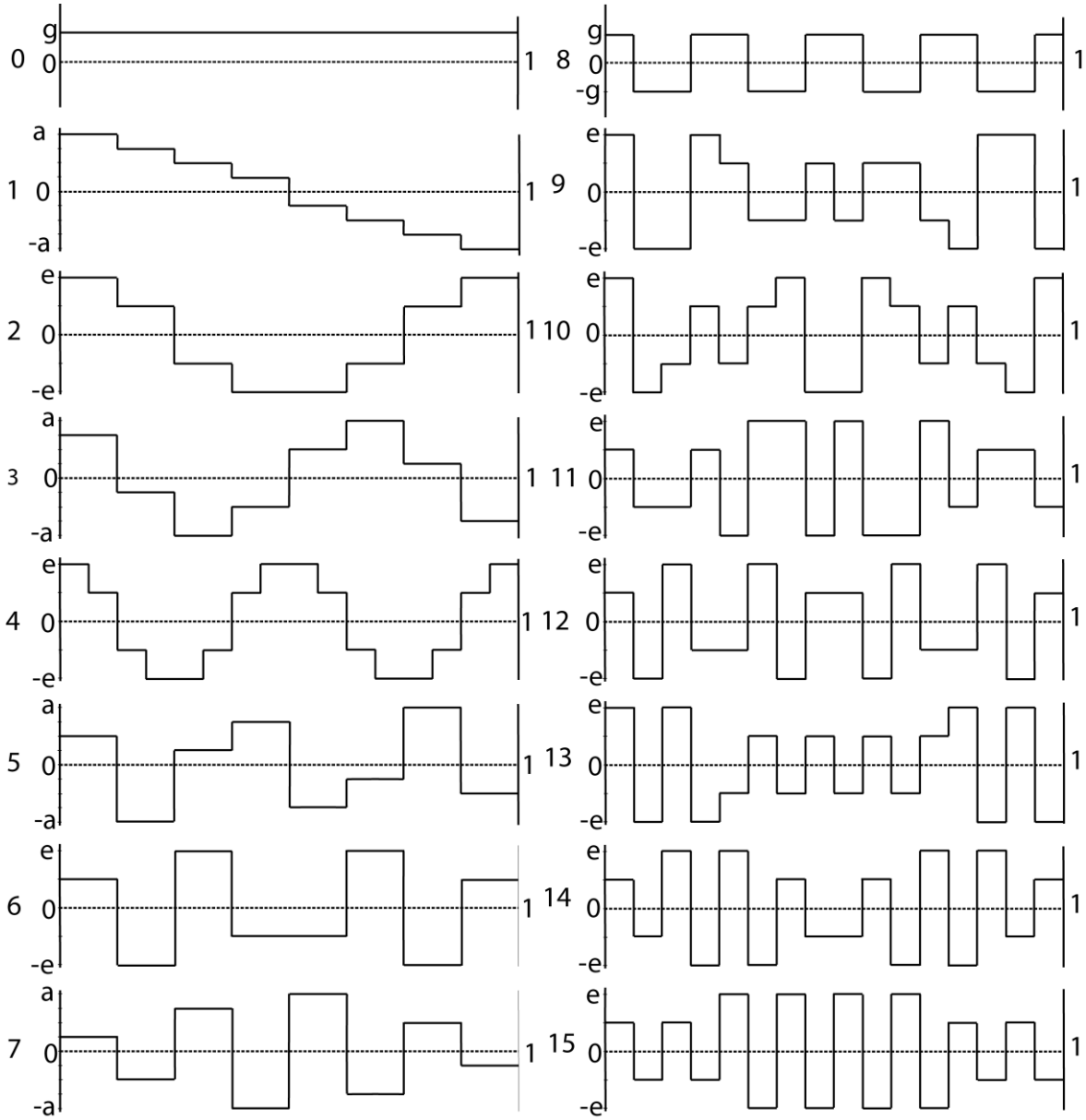

Рисунок 12. 16 базисних функцій простого ЦКП

Матриця ICT_{16} може бути представлена через матрицю ядра простого цілочислового косинусного перетворення:

$$ICT_{16} = B_{16}C_{16}, \tag{1}$$

де ICT_{16} – матриця 16x16 простого ЦКП, C_{16} – матриця ядра простого ЦКП, B_{16} – діагональна матриця 16x16 коефіцієнтів нормування.

Тоді матриця C_{16}^{*} може бути представлена рекурентно на основі методу, запропонованого в [39]:

$$C_{16}^{*} = diag\left[S_8^{*}, Q_8\right] H_{16}^{*} \qquad (2)$$

$$S_8^{*} = G_8 P_8 S_8,$$

де C_{16}^{*} – матриця 16x16 ядра простого ЦКП з переставленими рядками, S_8 – матриця 8x8 ядра ЦСП, яке вперше було введено Л.О. Гнатівим в [39], і являється узагальненням цілочисельного похилого перетворення в базисі ступінчастих функцій [38], S_8^{*} – матриця 8x8 ядра ЦСП з переставленими рядками на основі двійково-інверсних перестановок (ДІП) і перестановок по коду Грея (ПКГ), Q_8 – матриця 8x8, яка містить цілі елементи $\pm a$, $\pm b$, $\pm c$, $\pm d$, $\pm e$ і $\pm f$, P_8, S_8 – матриці 8x8 ДІП і ПКГ відповідно.

Матриця Q_8 може бути представлена рекурентно як добуток трьох матриць:

$$Q_8 = diag\left[Q_4, T_4\right] H_8' \, P_8^{T}, \; H_8' = H_2' \otimes I_4, \; H_2' = \begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \qquad (3)$$

де H_8' – фактор-матриця 8x8 з ненульовими елементами ± 1, P_8^{T} – матриця 8x8 обернених досконалих перестановок [1], Q_4 – матриця 4x4 , яка містить цілі елементи $\pm a$, $\pm b$, $\pm c$, $\pm d$; T_4 – матриця 4x4, яка містить цілі елементи $\pm e$ і $\pm f$, $\otimes$– операція кронекерівського добутку матриць, I_4 – одинична діагональна матриця 4x4.

Матриця T_4 може бути представлена через матрицю цілочисельного ступінчастого перетворення типу IV (ЦСП-IV) :

$$T_4 = I_4^* \tilde{S}_4^{IV},\ \tilde{S}_4^{IV} = \overline{I}_4 G_4 P_4 S_4, \qquad (4)$$
$$G_4 = diag\left[I_2, \overline{I}_2\right],\ P_4 = diag\left[1, \overline{I}_2, 1\right],\ I_4^* = diag\left[-I_2, I_2\right],$$

де $\tilde{S}_4^{IV}$ – матриця 4х4 ядра ЦСП-IV (аналог ядра ЦКП-IV) з переставленими рядками, P_4, G_4 – матриці 4х4 ДІП і ПКГ відповідно, $\overline{I}_4$ – одинична антидіагональна матриця 4х4, I_2 – одинична діагональна матриця 2х2.

Матриця Q_4 може бути представлена через матрицю ядра ЦКП-IV:

$$Q_4 = C_4^{IV} \overline{I}_4, \qquad (5)$$

де C_4^{IV} – матриця 4х4 ядра ЦКП-IV.

Матриці $\tilde{S}_4^{IV}$, H_8', S_8 і Q_4 мають вигляд:

$$\tilde{S}_4^{IV} = \begin{bmatrix} f & f & -e & e \\ f & -f & e & -e \\ e & -e & -f & f \\ e & e & f & f \end{bmatrix}, \quad H_8' = \begin{bmatrix} I_4 & I_4 \\ -I_4 & I_4 \end{bmatrix},$$

$$S_8 = \begin{bmatrix} g & g & g & g & g & g & g & g \\ e & e & f & f & -f & -f & -e & -e \\ e & f & -f & -e & -e & -f & f & e \\ f & f & -e & -e & e & e & -f & -f \\ g & -g & -g & g & g & -g & -g & g \\ e & -e & -f & f & -f & f & e & -e \\ f & -e & e & -f & -f & e & -e & f \\ f & -f & e & -e & e & -e & f & -f \end{bmatrix}, \quad Q_4 = \begin{bmatrix} d & c & b & a \\ -c & -a & -d & b \\ b & d & -a & c \\ -a & b & -c & d \end{bmatrix},$$

(6)

Матриця Q_4 може бути представлена як добуток двох матриць:

$$Q_4 = R_4 M_4,$$

де R_4 – фактор-матриця 4x4, яка містить на головній діагоналі елементи r, а на протилежній $\pm s$, M_4 – фактор-матриця 4x4 з елементами $0, \pm 1, q$.

Тоді матриця Q_8 згідно (3) і на основі (4) та (5) може бути представлена наступним чином:

$$Q_8 = diag\left[C_4^{IV}\bar{I}_4, I_4^{*}\tilde{S}_4^{IV}\right] H_8' P_8^T. \qquad (7)$$

Блок-схема швидкого простого ЦКП порядку 16 на основі методу згідно (2) та враховуючи (7), представлена на рис. 13.

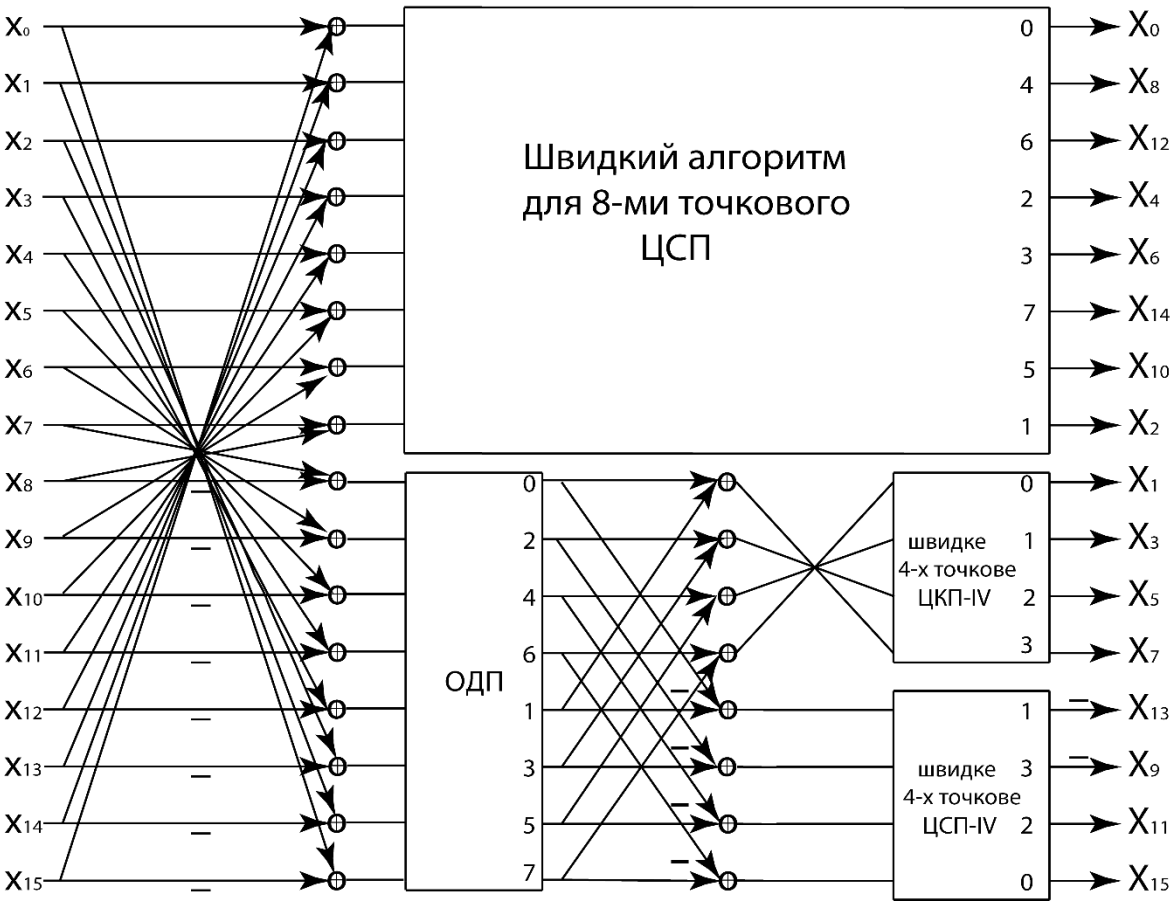

Рис. 13. Блок-схема швидкого простого ЦКП порядку 16.

Розглянемо алгоритм швидкого обчислення прямого цілочислового ступінчастого перетворення порядку 8, який є складовою частиною ЦКП порядку 16.

3.1.2 Алгоритм швидкого прямого цілочислового ступінчастого перетворення порядку 8

Матриця S_8^* може бути представлена як добуток трьох матриць:

$$S_8^* = S_3 S_2 S_1, \tag{8}$$

де S_i – i-ті ($i = 1,2,3$) фактор-матриці 8x8 алгоритму швидкого прямого ЦСП:

$$S_1 = H_8^*,\ S_2 = diag\left[H_4^*, S_4'\right],$$

$$S_3 = diag\left[T_2, S_2', H_2, \bar{H}_2\right]$$

$$H_4^* = \begin{vmatrix} 1 & & & 1 \\ & 1 & 1 & \\ & 1 & -1 & \\ 1 & & & -1 \end{vmatrix},\ S_4' = \begin{bmatrix} -e & & & f \\ & -e & f & \\ & f & e & \\ f & & & e \end{bmatrix},$$

$$T_2 = \begin{bmatrix} g & g \\ g & g \end{bmatrix},\ S_2' = \begin{bmatrix} -e & f \\ f & e \end{bmatrix},\ \bar{H}_2 = \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}.$$

На рис. 14 представлений граф алгоритму згідно (8) швидкого прямого ЦСП порядку 8.

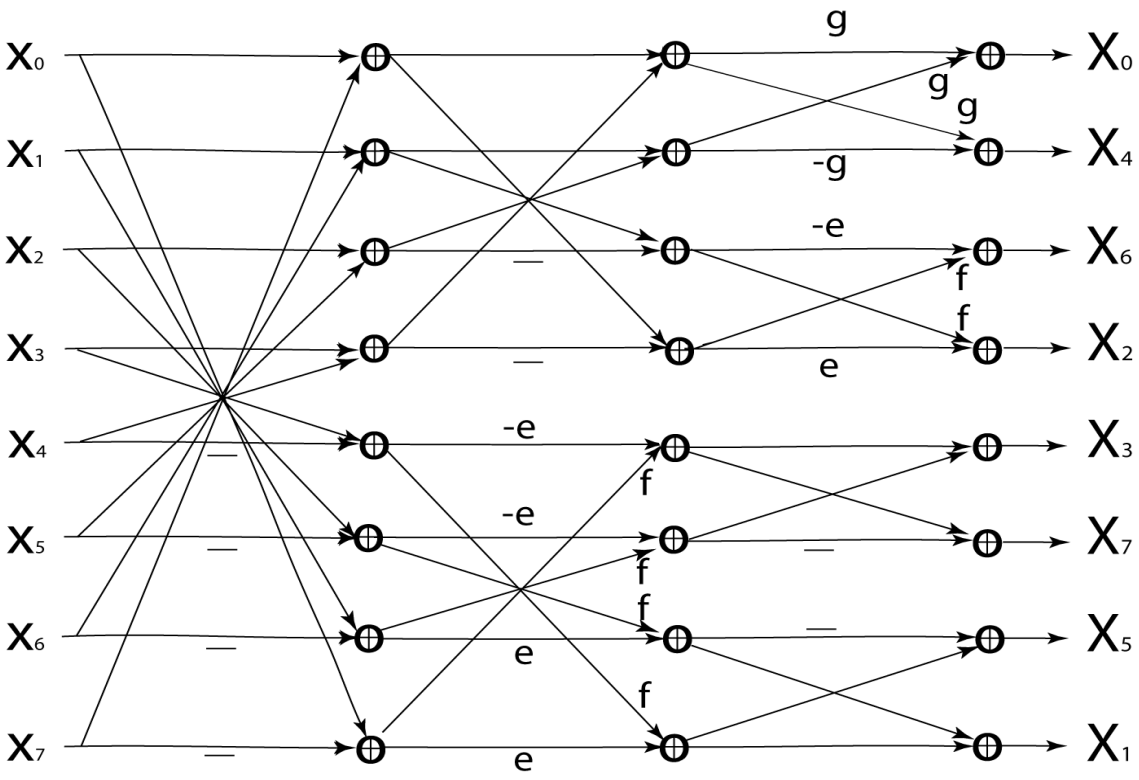

Рисунок 14. Граф алгоритму швидкого прямого ЦСП порядку 8

3.1.3 Алгоритм швидкого оберненого цілочислового ступінчастого перетворення порядку 8

Обернене ЦСП можна отримати шляхом транспонування матриці:

$$S_{8i} = S_8^{*T}, \tag{9}$$

де S_{8i} – матриця оберненого ЦСП.

Матриця S_{8i} на основі (9) і алгоритму швидкого прямого ЦСП згідно (8) та враховуючи симетричність фактор-матриць ($S_1' = S_1$, $S_2'^T = S_2'$, $\bar{H}_2^T = \bar{H}_2$, $H_8^{*T} = H_8^*$), може бути представлена як добуток трьох матриць

$$S_{8i} = S_1 S_2 S_3 \tag{10}$$

де S_i – i-ті ($i=1,2,3$) фактор-матриці 8х8 алгоритму швидкого оберненого ЦСП.

Граф алгоритму швидкого оберненого ЦСП порядку 8 згідно (10) представлений на рис. 15.

На основі ядра ЦКП розмірності 8, що задано як $C_8[a,b,c,d;e,f;g]$ із цілими елементами, які для прямого перетворення приймають значення $C_8[5,3,2,1;4,2;4]$, а для оберненого – $C_8[5,3,2,1;4,2;4]/4$, за наведеною схемою можна побудувати нове просте ЦКП порядку 16, матриця ядра T_1 якого представлена нижче.

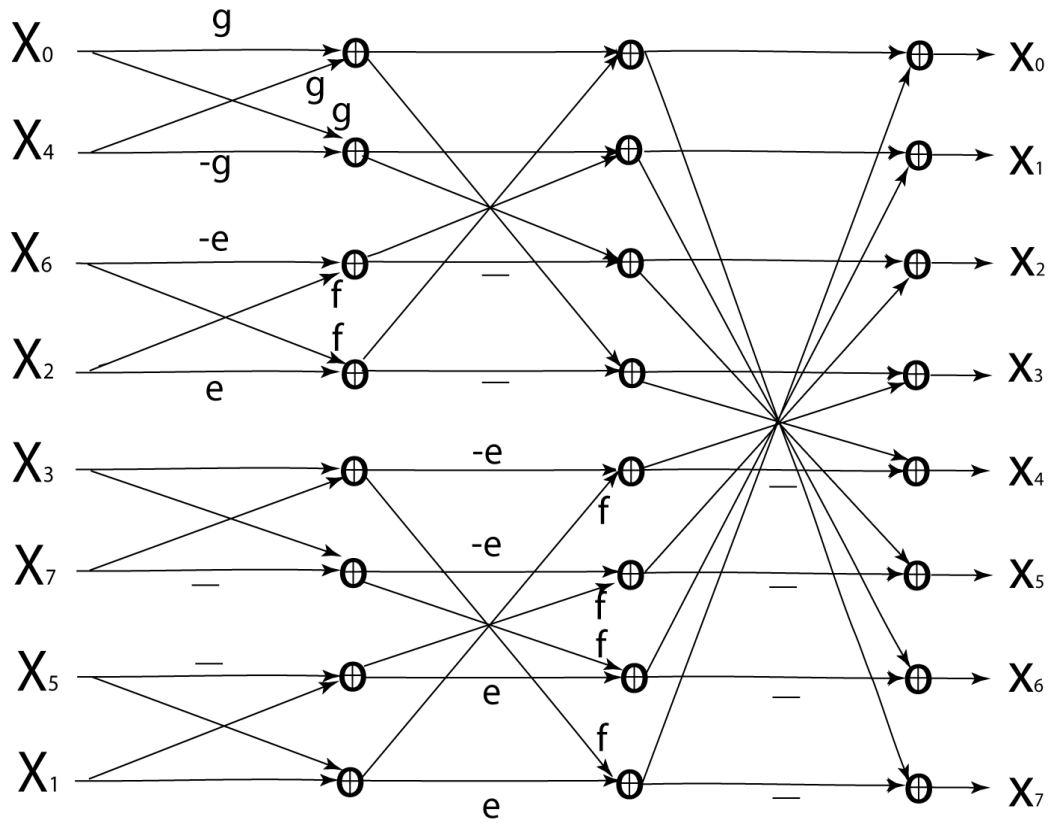

Рисунок 15. Граф алгоритму швидкого оберненого ЦСП порядку 8

$$T_1 = \begin{vmatrix}
4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \\
5 & 5 & 3 & 3 & 2 & 2 & 1 & 1 & -1 & -1 & -2 & -2 & -3 & -3 & -5 & -5 \\
4 & 4 & 2 & 2 & -2 & -2 & -4 & -4 & -4 & -4 & -2 & -2 & 2 & 2 & 4 & 4 \\
3 & 3 & -1 & -1 & -5 & -5 & -2 & -2 & 2 & 2 & 5 & 5 & 1 & 1 & -3 & -3 \\
4 & 2 & -2 & -4 & -4 & -2 & 2 & 4 & 4 & 2 & -2 & -4 & -4 & -2 & 2 & 4 \\
2 & 2 & -5 & -5 & 1 & 1 & 3 & 3 & -3 & -3 & -1 & -1 & 5 & 5 & -2 & -2 \\
2 & 2 & -4 & -4 & 4 & 4 & -2 & -2 & -2 & -2 & 4 & 4 & -4 & -4 & 2 & 2 \\
1 & 1 & -2 & -2 & 3 & 3 & -5 & -5 & 5 & 5 & -3 & -3 & 2 & 2 & -1 & -1 \\
4 & -4 & -4 & 4 & 4 & -4 & -4 & 4 & 4 & -4 & -4 & 4 & 4 & -4 & -4 & 4 \\
4 & -4 & -4 & 4 & 2 & -2 & -2 & 2 & -2 & 2 & 2 & -2 & -4 & 4 & 4 & -4 \\
4 & -4 & -2 & 2 & -2 & 2 & 4 & -4 & -4 & 4 & 2 & -2 & 2 & -2 & -4 & 4 \\
2 & -2 & -2 & 2 & -4 & 4 & 4 & -4 & 4 & -4 & -4 & 4 & -2 & 2 & 2 & -2 \\
2 & -4 & 4 & -2 & -2 & 4 & -4 & 2 & 2 & -4 & 4 & -2 & -2 & 4 & -4 & 2 \\
4 & -4 & 4 & -4 & -2 & 2 & -2 & 2 & -2 & 2 & -2 & 2 & 4 & -4 & 4 & -4 \\
2 & -2 & 4 & -4 & 4 & -4 & 2 & -2 & -2 & 2 & -4 & 4 & -4 & 4 & -2 & 2 \\
2 & -2 & 2 & -2 & 4 & -4 & 4 & -4 & 4 & -4 & 4 & -4 & 2 & -2 & 2 & -2
\end{vmatrix}.$$

Ще одним критерієм для оцінки перетворень є динамічний діапазон чисел для представлення елементів матриці перетворення. Для наведеної матриці простого ЦКП достатньо лише 3 бітів, в той час як для елементів матриць перетворень, застосованих в стандарті H.265, необхідно вже 7 біт.

Нескладно побудувати й просте ЦКП порядку 16, використовуючи числа із стандарту H.265, тобто використавши за основу ядро ЦКП розмірності 8 з елементами $C_8\left[89,75,50,18;83,36;64\right]$. Але для нього обчислювальна складність буде вищою, і включати операції множення, при тому ж рівні ефективності.

3.1.4 Обчислювальна складність цілочислового простого косинусного перетворення порядку 16

Запропоновані перетворення можуть використовуватись в трьох формах: матричне множення, часткова факторизація («partial butterfly», PB) и повна факторизація («full-factorization form», FF). У табл. 11 наведена обчислювальна

складність двох вищезгаданих перетворень, а для порівняння наведена обчислювальна складність косинусного перетворення розмірності 16, яке використовується у стандарті H.265 [27].

При обчисленні загальної кількості операцій замінимо 1 множення на 2 додавання, по мінімуму (хоча в цілому одне множення зазвичай заміняється та виконується більшою кількістю додавань й зсувів).

Але навіть без врахування типу операцій, загальна кількість операцій для запропонованого простого ЦКП буде в 2,2 рази меншою, ніж для H.265, а з урахуванням – в 3,2 рази меншою (за оцінкою знизу).

Таблиця 11.

Кількість операцій	T_1	$T_{\text{H.265}}$	H.265
множення	–	36	86
додавання	72	68	100
зсуву	18	16	16
Кількість ітерацій	4	4	4
нормування	+	–	–
Кількість біт елемента матриці	3	7	7

3.1.5 Тестування і порівняльний аналіз цілочислового простого косинусного перетворення порядку 16

Наведемо результати тестування запропонованого перетворення з урахуванням алгоритмів передбачення значень пікселів, що дає більш точну картину відносної ефективності перетворень, оскільки без передбачення пікселів перетворення застосовувати недоцільно (в статті [38] алгоритми передбачення не враховувались, що дало недостатньо точні результати порівняння

перетворень). В якості алгоритму передбачення застосуємо DC передбачення за середнім значенням сусіднього блоку.

Для зображення «храм» (рисунок 9), що має середню кореляцію пікселів, тобто належить до найбільш статистично значимої групи зображень, результати за характеристикою PSNR наведено в табл. 12, а різниці між перетвореннями за PSNR – в табл. 13. Як бачимо, різниці складають від 0,17 до 0,26 дБ., що вкладається у візуально непомітний діапазон спотворень до 0,5 дБ.

Таблиця 12

Перетворення	Qpst	PSNR
Запропоноване ЦПСП-16	8	42.03
	14	37.71
	24	33.96
HEVC-16	8	42.20
	14	37.95
	24	34.22

Таблиця 13

Qpst	Різниця PSNR
8	0.17
14	0.24
24	0.26

Для зображень «фото» і «пейзаж» (рис. 10 і 11), що мають високу і низьку кореляцію пікселів, результати показані в табл. 14 і 15.

Таблиця 14

Перетворення	Qpst	PSNR
Запропоноване ЦПКП-16	8	41.44
	14	38.37
	24	36.32
HEVC-16	8	41.53
	14	38.46
	24	36.46

Різниці за характеристикою PSNR для «фото» коливаються від 0,09 до 0,12 дБ., тобто ще менші, ніж для зображення «храм», а для зображення «пейзаж» становлять 0,07-0,26 дБ., причому на більших Qpst різниці менші за 0,1 дБ.

Таблиця 15

Перетворення	Qpst	PSNR
Запропоноване ЦПКП-16	8	40.60
	14	36.08
	24	31.69
HEVC-16	8	40.86
	14	36.17
	24	31.76

Таким чином, запропоноване цілочислове просте косинусне перетворення порядку 16 зі швидкими алгоритмами, що не потребують операцій множення, має достатньо високі показники якості, щоб використовуватись для збільшення швидкодії кодування на середніх і великих параметрах квантування. Але коефіцієнт стиснення у простого косинусного перетворення трохи менший, ніж у традиційного косинусного, оскільки половина функцій є ступінчастими, які мають гірші показники акумуляції інформації та енергії в найбільших коефіцієнтах. Зробимо оцінку коефіцієнта стиснення на трьох тестових зображеннях, що представляють три різні групи зображень (низької, середньої і високої кореляції даних), а саме «фото», «храм» і «пейзаж» (табл. 16).

Таблиця 16

Зображення	QPst	Розмір, кб для ПЦКП	Розмір, кб для ЦКП	Збільшення *K* стиснення, %
Фото	8	125	123	1,6
	14	67.5	66.0	3,0
	24	35.0	34.1	2,9
Храм	8	185	176	5,1
	14	139	132	5,3
	24	101	94.7	6,3
Пейзаж	8	289	280	3,2
	14	225	219	2,7
	24	170	165	3,0

Як бачимо, різниця в середньому складає 3%, з варіацією до близько 5%.

Розглянемо інший варіант передбачення – на основі зовнішніх даних, тобто в рамках підходу прогресивної передачі даних. Візьмемо за передбачуване значення блоку розміром 4x4 його середнє значення.

В результаті отримаємо менші різниці, результат передбачення, ніж для випадку передбачення для блоку 16x16 на основі значень сусідніх пікселів зображення. Результати тестування двох перетворень наведено в табл. 17.

Таблиця 17

Перетворення	QPst	Розмір, кб	PSNR
HEVC-16	8	226	42.21
	14	177	37.69
	24	137	33.91
Запропоноване ЦПКП-16	8	225	42,17
	14	178	37,66
	24	137	33,85

Як бачимо, різниця між перетвореннями за коефіцієнтом стиснення і за характеристикою PSNR значно зменшилась. Причина цього лежить в зменшенні ваги перетворення, тобто більша частина стиснення була отримана на етапі передбачення, тому ефективність перетворення отриманих після передбачення різниць відіграє значно меншу роль.

Але потрібно пам’ятати, що зменшене зображення найменшого масштабу може використовувати алгоритми передбачення лише на основі сусідніх блоків, тому для нього ефективність перетворення має більшу вагу.

Відповідно, для кодування різниць між зображеннями різного масштабу краще підходить просте косинусне перетворення порядку 16 внаслідок його меншої обчислювальної складності, а для кодування зображення з мінімальним масштабом – косинусне перетворення порядку 16, враховуючи його більшу на 3-5% ефективність стиснення.

Таким чином, ми ще раз переконались, що використання зображень різних масштабів потрібно не тільки для прогресивної передачі даних, але й для

зменшення обчислювальної складності кодування, а саме – зменшення кількості алгоритмів передбачення, і використання спрощених перетворень.

3.2 Цілочислове синусне перетворення типу VII порядку 4

Джайн [76, 77] показав, що ПКЛ для залишкових процесів буде ДСП, коли граничні умови є доступними в обох напрямках. Для покращення кодування з перетворенням прогнозованих блоків в моді intra в [69] було введено моде-залежне направлене перетворення (“mode-dependent directional transform”, MDDT) яке є ядром тестової моделі TMuC [78] стандарту відеокодування H.265/HEVC [27].

В [79] для апроксимації роздільного направленного перетворення [69] було використано тільки дві матриці перетворення: ДКП і ДСП. Було показано, що залишки, отримані в результаті intra прогнозування при кодуванні зображень/відео представляють сепарабельну корреляційну модель зображення з коефіцієнтами кореляції між сусідніми пікселями у вертикальній та горизонтальній проекціях.

Відмічено, що ПКЛ коваріаційної Теплицевої матриці, отриманої з рядка залишку після intra прогнозування, є наближено ДКП. Оскільки коваріаційна матриця, отримана зі стовпця залишку, не є Теплицевою матрицею, то ДКП є лише підоптимальним (субоптимальним) перетворенням. Оптимальним перетворенням (ПКЛ), яке приводить цю коваріаційну матрицю до діагонального вигляду, є трьохдіагональна матриця, власні вектора якої можуть бути обчислені [80] як дискретне синусне перетворення типу VII (ДСП-VII). Аналогічний результат аналітично отримано в [81] для гаусово-марківського процесу першого порядку, де показано, що ПКЛ для цього процесу є ДСП-VII при коефіцієнті кореляції ρ, що прямує до нуля, і де було вперше опубліковано модозалежне ДКП/ДСП. Зауважимо, що згідно вище запропонованої мат.моделі, залишки після передбачення мають дві складові: хаотичну і закономірну.

У пропозиції [82] до стандарту HEVC отримано цілочислові апроксимації матричного обчислення ДСП-VII порядку $N = 4,8,16$, а також приведено алгоритм 4-точкового швидкого обчислення цілочислового синусного перетворення типу VII (ЦСП-VII), який не є оптимальним за мультиплікативною складністю, оскільки потребує 8 операцій множення.

Розглянемо метод побудови цілочислового синусного перетворення типу VII порядку 4.

Матриця N-точкового ДСП типу VII може бути визначена як [80, 83]:

$$\left[S_N^{VII}\right]_{k,n} = \frac{2}{\sqrt{2N+1}} \sin\left[\frac{(2k+1)(n+1)\pi}{2N+1}\right],\ k,n = 0,1,...,N-1.$$

У роботі [84] наведено три способи представлення матриці S_N^{VII} ДСП-VII через уявну частину 2N+1-точкового дискретного перетворення Фур'є, а також приведені алгоритми швидкого обчислення ДСП-VII порядку 4 і 8.

В [82, 85] приведено цілочислові апроксимації матричного обчислення ДСП-VII, які для порядку N=4 співпадають, а для N=8, 16 різні.

Розглянемо матрицю IST_4^{VII*} розміру 4x4 цілочислового синусного перетворення типу VII (ЦСП-VII) з переставленими рядками і стовпцями, яка отримана з матриці IST_4^{VII} перестановкою рядків на основі простих досконалих перестановок (ПДП) і перестановкою стовпців на основі двійково-інверсних перестановок (ДІП) [86]:

$$IST_4^{VII*} = \tilde{P}_4 IST_4^{VII} P_4, \qquad (11)$$

де P_4 - матриця 4x4 ПСП, $\tilde{P}_4 = antidiag\left[I_2, I_2\right]$, $\tilde{P}_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$.

Матриця IST_4^{VII*} розміру 4x4 ЦСП-VII з переставленими рядками і стовпцями може бути представлена матрицею ядра ЦСП-VII:

$$IST_4^{VII*} = B_4 S_4^{VII*}, \tag{12}$$

де S_4^{VII*} – матриця 4x4 ядра ЦСП-VII з переставленими рядками і стовпцями, B_4 – діагональна матриця 4x4 коефіцієнтів нормування.

Матриця S_4^{VII*} містить цілі елементи $\pm a, b, \pm c, \pm d$ і нуль та має наступний вигляд:

$$S_4^{VII*} = \begin{bmatrix} d & -c & -a* & b \\ b* & c & -d & -a \\ a & c & b & d* \\ c & 0 & c & -c \end{bmatrix}, \tag{13}$$

де $a < b < c < d$, $-a^* = b - d$, $b^* = d - a$, $d^* = a + b$. Тут матрицю S_4^{VII*} можна представити прямокутною матрицею $S_{3,4}^{III*}$ розміру 3x4 з вектор-рядком $S_{1,4}^*$ розмірності 4. При цьому

$$S_{3,4}^{III*} = \begin{bmatrix} d & -c & -a^* & b \\ b^* & c & -d & -a \\ a & c & b & d^* \end{bmatrix}, \; S_{1,4}^* = (c, 0, c, -c). \tag{14}$$

Матрицю $S_{3,4}^{III*}$ можна представити добутком двох фактор-матриць:

$$S_{3,4}^{III*} = S_2 S_1, \tag{15}$$

де S_1, S_2 – фактор-матриці розміру 4x4 і 3x4 алгоритма, запропонованого в [87] швидкого виконання 4-точкового прямого ЦСП-VII: $S_1 = T_4$, $S_2 = T_{3,4}$.

Фактор-матриця 4х4 T_4 містить ненульові елементи ± 1.

Матрицю $T_{3,4}$ можна представити як добуток двох матриць:

$$T_{3,4} = M_{3,4} R_4, \tag{16}$$

де R_4 – діагональна матриця 4х4 з елементами d, c, a, b, $M_{3,4}$ – фактор-матриця 3х4, яка містить в кожному рядку по три ненульових елементи ± 1,

$$R_4 = diag\left[d, c, a, b\right]. \tag{17}$$

Алгоритм швидкого обчислення 4-точкового ЦСП-VII виконується за три ітерації. Перша ітерація полягає у множенні фактор-матриці T_4 (15), що містить у трьох рядках по два ненульові елементи ± 1, на вхідний вектор-стовпець x_4 розмірності 4, що потребує трьох операцій додавання.

На другій ітерації виконується множення отриманих результатів першої ітерації на цілі елементи d, c, a, b згідно діагональної матриці R_4 (17), що потребує 4 операції множення.

На третій ітерації, яка полягає у множенні прямокутної фактор-матриці $M_{3,4}$, що містить у трьох рядках по три ненульових елементи ± 1, на вектор-стовпець X_4 результатів другої ітерації, що потребує шість операцій додавання. При цьому також виконується операція множення вектора рядка $S^*_{1,4}$, який згідно (14) містить елементи $\pm c$ і нуль на вхідний вектор-стовпець x_4 , що потребує дві операції додавання і однієї операції множення на елемент c. Всього алгоритм потребує 5 операцій множення і 11 операцій додавання.

Алгоритм швидкого обчислення 4-точкового оберненого ЦСП-VII.

Матрицю $\left(S_4^{VII}\right)^{-1}$ оберненого ЦСП-VII порядку 4 можна отримати транспонуванням:

$$\left(S_4^{VII}\right)^{-1} = S_4^{VII*T}. \tag{18}$$

Матриця S_4^{VII*T} має вигляд:

$$S_4^{VII*T} = \begin{bmatrix} d & b^* & a & c \\ -c & c & c & 0 \\ -a^* & -d & b & c \\ b & -a & d^* & -c \end{bmatrix}. \tag{19}$$

Матрицю S_4^{VII*T} можна представити прямокутною матрицею $S_{3,4}^{VII*T}$ розміру 3x4 і вектор-рядком $S_{1,4}^{*T}$ розмірності 4.

При цьому

$$S_{3,4}^{VII*T} = \begin{bmatrix} d & b^* & a & c \\ -a^* & -d & b & c \\ b & -a & d^* & -c \end{bmatrix},\ S_{1,4}^{*T} = \left(-c, c, c, 0\right). \tag{20}$$

Матриця $S_{3,4}^{VII*T}$ можна представити як добуток двох фактор-матриць:

$$S_{3,4}^{VII*T} = S_2^{-1} S_1^{-1}, \tag{21}$$

де S_1^{-1}, S_2^{-1} – обернені фактор-матриці алгоритму швидкого виконання 4-точкового оберненого ЦСП-VII:

$$S_1^{-1} = T_4^{-1},\ S_2^{-1} = T_{3,4}^{-1}. \tag{22}$$

При цьому обернена фактор-матриця T_4^{-1} розміру 4x4 містить у трьох рядках по два ненульові елементи ± 1, а обернена фактор-матриця $T_{3,4}{}^{-1}$ розміру 3x4 містить ненульові елементи $a, b, \pm c, \pm d$, і яку можна представити добутком двох матриць:

$$T_{3,4}^{-1} = M_{3,4}^{-1} R_4, \tag{23}$$

де R_4 – діагональна матриця (17), $M_{3,4}^{-1}$ – обернена фактор-матриця 3x4, яка містить в кожному рядку по три ненульових елемента ± 1.

Таким чином, алгоритм швидкого обчислення 4-точкового оберненого ЦСП-VII згідно (20)-(23) аналогічно алгоритму швидкого обчислення прямого ЦСП-VII потребує 5 операцій множення і 11 операцій додавання.

Матриця S_4^{VII} ядра ЦСП-VII порядка 4 має вигляд:

$$S_4^{VII} = \begin{bmatrix} a & b & c & d \\ d & c & 0 & -c \\ d & -a & -c & b \\ b & -d & c & -a \end{bmatrix}, \quad a+b=d \tag{24}$$

Елементи матриці S_4^{VII} однонормового масштабованого ЦСП-VII-1, із роботи [87] приймають наступні значення: $a=14$, $b=28$, $c=37, d=42$. В стандарті H.265 елементи матриці S_4^{VII} приймають наступні значення: a=29, b=55, c=74, d=84. В [88] приведено цілочислову апроксимацію із заміною елементів матриці S_4^{VII}: a=28, b=56 для зменшення операцій множення у алгоритмі швидкого виконання 4-точкового ЦСП-VII.

Матриця S_4^{VII} запропонованого однонормового масштабованого ЦСП-1 має вигляд:

$$S_4^{VII} - 1 = \begin{bmatrix} 14 & 28 & 37 & 42 \\ 37 & 37 & 0 & -37 \\ 42 & -14 & -37 & 28 \\ 28 & -42 & 37 & -14 \end{bmatrix}. \tag{25}$$

Базисні вектори матриці $S_4^{VII} - 1$ масштабованого ЦСП-1 мають L_2 норму, яка наближається до числа степені два: $\|S_i\|^2 = 4096 + \Delta_i$, де $\|S_i\|^2$ – L_2-норма i-ої базисної функції матриці перетворення (яка прямує до одиниці для перетворення

з ортонормованим базисом), Δ_i (відсотки, %) – відхилення L_2-норми i-ої базисної функції становить Δ_i=0,42%, $i=0,2,3$, Δ_1=0,27%, неортогональність складає 0,07%. У роботі [87] запропоновано інше однонормове масштабоване ЦСП-2, елементи матриці якого представляються шістьма бітами, а базисні вектори мають однакову L_2-норму, яка представляється трьохрозрядним числом, а неортогональність становить 0,5 %.

Таким чином ЦСП-1 є близьким до ДСП-VII і має такі ж властивості, як і ЦСП-VII, прийняте в стандарті Н.265: масштабоване перетворення та його коефіцієнти є достатньо близькими до ДСП-VII; неортогональність базисних векторів в результаті проведеного аналізу складає 0,07% і 0,5% (майже ортогональні); однакова схема квантування і деквантування, де множники залежать від значень параметру квантування QP [6].

Обчислювальна складність запропонованого алгоритму ЦСП-1, відомого алгоритму ЦСП [79], відомого алгоритму ЦСП [88] і відомого алгоритму ДСП [83] 1D 4-точкових перетворень наведена у табл. 18.

Таблиця 18

Операції	Оцінка обчислювальної складності 1D 4-точкових перетворень			
	Запропонованих	Відомих		
	ЦСП-1	ЦСП (Н.265) [79]	ЦСП [88]	ДСП [83]
Множення	5	8	5	5
Додавання + Зсув	11	10	11+3	11

Обчислювальна складність запропонованих роздільних адаптивних перетворень цілочислового косинусного перетворення (ЦКП) [40]+ЦСП-1, відомого [89] і прийнятих у стандарті Н.265 ЦКП [90]+ЦСП [79] 2D 4-точкових цілочислових обернених перетворень наведена у табл. 19.

Таблиця 19

Операції	Оцінка обчислювальної складності 2D роздільних обернених перетворень для блоків 4x4			Порівняльний аналіз ЦКП+ЦСП-1 відносно	
	Запропонованого ЦКП+ЦСП-1	Відомих ЦКП+ЦСП		H.265[90]+ [79]	[89]
	ЦСП-1	H.265[90] + [79]	із [89]		
Множення	20	56	56	В 2,8 рази менше	В 2,8 рази менше
Додавання+ зсув	100+24	72+16	108+32	На 40,9 % більше додавань	На 11,4 % менше додавань

Алгоритм запропонованого ЦСП-1 потребує на 37,5% менше операцій множення у порівнянні з відомим алгоритмом [79].

3.3 Цілочислове однонормове модифіковане синус-косинусне перетворення порядку 8 типу VII

Роздільні адаптивні перетворення були запропоновані для спільного застосування косинусних і синусних перетворень [69], коли замість традиційного 2D косинусного перетворення різних порядків емпірично порівнюються і обираються найкраща комбінація 1D косинусного і синусного перетворень з чотирьох можливих комбінацій. В H.265 використовувались лише роздільні адаптивні перетворення порядку 4, оскільки синусні перетворення більших порядків мають велику обчислювальну складність. Але в H.266 застосували і синусні перетворення більших порядків, включаючи 16.

В якості альтернативи синусним перетворенням пропонується використовувати синус-косинусні перетворення. Зокрема, з цією метою розроблено цілочислове однонормове модифіковане синус-косинусне перетворення порядку 8 зі швидкими алгоритмами обчислення низької обчислювальної складності.

3.3.1 Метод побудови цілочислового модифікованого синус-косинусного перетворення порядку 8 типу VII

У пропозиціях [79, 85] наведено дві різні однонормові цілочисельні апроксимації матричного обчислення ДСП-VII порядку 8 і 16. У [83] наведено три способи представлення матриці S_N^{VII} N-точкового ДСП-VII через уявну частину $2N+1$-точкового дискретного перетворення Фур'є.

Матрицю N-точкового ДСП-VII (також відомого як «непарне» (odd) перетворення типу III [91] або типу I [92]) можна визначити як

$$\left[S_N^{VII}\right]_{k,n} = \frac{2}{\sqrt{2N+1}} \sin\left[\frac{(2k+1)(n+1)\pi}{2N+1}\right],\ k,n = 0,1,...,N-1. \qquad (26)$$

У роботі [16] побудовано матрицю цілочислового модифікованого синус-косинусного перетворення типу VII (ЦМСКП-VII) порядку 8, в якій рядки з парними номерами представляють базисні синусні функції ДСП-VII, симетричні відносно середини інтервалу. Ці функції названі парними модифікованими синусними функціями. Рядки з непарними номерами представляють базисні функції ДКП, антисиметричні відносно середини інтервалу. Ці функції названі непарними модифікованими косинусними функціями [16].

Розглянемо матрицю $IMSCT_8^{VII*}$ розміру 8x8 ЦМСКП-VII з переставленими рядками, отриману шляхом перестановки рядків на основі обернених досконалих перестановок (ОДП), простих досконалих перестановок (ПДП) і двійково-інверсних перестановок (ДІП) [86]:

$$IMSCT_8^{VII*} = \tilde{P}_8 P_8 IMSCT_8^{VII}\ , \qquad (27)$$

де P_8 — матриця 8x8 ОДП, $P_8(0,7) = (0,2,4,6,1,3,5,7)$, $\tilde{P}_8$ — блочно-діагональна матриця 8x8, яка містить матрицю розміру 4x4 $\bar{P}_4$ ПДП і матрицю розміру 4x4

ДІП, $\tilde{P}_8 = diag\left(\bar{P}_4, P_4\right)$, $\bar{P}_4 = antidiag\left[I_2, I_2\right]$ — антидіагональна матриця 4х4, яка містить одиничні матриці I_2 розміру 2х2, $P_4 = diag\left(1, \bar{I}_2, 1\right)$ — діагональна матриця 4х4, яка містить елементи 1 і антидіагональну одиничну матрицю $\bar{I}_2$ розміру 2х2:

$$\bar{I}_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \bar{P}_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad P_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Матрицю $IMSCT_8^{VII*}$ розміру 8х8 ЦМСКП-VII з переставленими рядками можна записати через матрицю ядра ЦМСКП-VII:

$$IMSCT_8^{VII*} = B_8 MSC_8^{VII*}, \tag{28}$$

де MSC_8^{VII*} — матриця 8х8 ядра ЦМСКП-VII з переставленими рядками, B_8 — діагональна матриця 8х8 коефіцієнтів нормування.

Матрицю MSC_8^{VII*} можна побудувати з використанням рекурентного методу [16]:

$$MSC_8^{VII*} = diag\left[S_4^{VII*}, C_4^*\right] H_8^*, \tag{29}$$

де S_4^{VII*} — матриця 4х4 ядра ЦСП-VII з переставленими рядками на основі ПДП і переставленими стовпцями на основі ДІП, C_4^* — матриця 4х4 ядра ЦКП з переставленими рядками на основі ДІП:

$$S_4^{VII*} = \tilde{P}_4 S_4^{VII}, \; C_4^* = P_4 C_4, \tag{30}$$

S_4^{VII}, C_4 — матриці 4х4 ядер ЦСП-VII (2.20) і ЦКП відповідно, H_8^* — фактор-матриця 8х8 із ненульовими елементами ± 1,

$$H_8^* = \begin{bmatrix} I_4 & \bar{I}_4 \\ \bar{I}_4 & -I_4 \end{bmatrix},\ \bar{I}_4 = antidiagI_4,\ \bar{I}_4 = \begin{bmatrix} & & & 1 \\ & & 1 & \\ & 1 & & \\ 1 & & & \end{bmatrix}, \tag{31}$$

I_4, $\bar{I}_4$ — одинична та антидіагональна одинична матриці 4х4.

При цьому

$$C_4 = \begin{bmatrix} g & g & g & g \\ e & f & -f & -e \\ g & -g & -g & g \\ f & -e & e & -f \end{bmatrix}, \tag{32}$$

де $e > f, g = 2^m, m - ціле$.

Матриця MSC_8^{VII} ядра ЦМСКП-VII порядку 8 на основі (14) і (17) має наступний вигляд [16]

$$MSC_8^{VII} = \begin{bmatrix} a & b & c & d & d & c & b & a \\ g & g & g & g & -g & -g & -g & -g \\ c & c & 0 & -c & -c & 0 & c & c \\ e & f & -f & -e & e & f & -f & -e \\ d & -a & -c & b & b & -c & -a & d \\ g & -g & -g & g & -g & g & g & -g \\ b & -d & c & -a & -a & c & -d & b \\ f & -e & e & -f & f & -e & e & -f \end{bmatrix}, \tag{33}$$

Елементи матриці MSC_8^{VII} однонормового масштабованого модифікованого ЦСКП-VII-1 набувають таких значень: a=14, b=28, c=37, d=42, e=43, f=14, g=32.

Матриця MSC_8^{VII}-1 запропонованого масштабованого ЦСКП-VII-1 порядку 8 має вигляд [16]

$$MSC_8^{VII}-1=\begin{bmatrix} 14 & 28 & 37 & 42 & 42 & 37 & 28 & 14 \\ 32 & 32 & 32 & 32 & -32 & -32 & -32 & -32 \\ 37 & 37 & 0 & -37 & -37 & 0 & 37 & 37 \\ 43 & 14 & -14 & -43 & 43 & 14 & -14 & -43 \\ 42 & -14 & -37 & 28 & 28 & -37 & -14 & 42 \\ 32 & -32 & -32 & 32 & -32 & 32 & 32 & -32 \\ 28 & -42 & 37 & -14 & -14 & 37 & -42 & 28 \\ 14 & -43 & 43 & -14 & 14 & -43 & 43 & -14 \end{bmatrix}. \qquad (34)$$

Базисні вектори матриці MSC_8^{VII}-1 мають L_2-норму, яка наближається до степеня числа два: $\left\|S_i^{(1)}\right\|^2 = 8192 \pm \Delta_i$, де $\|S_i\|^2$ – L_2-норма i-ї базисної функції матриці перетворення (яка прямує до одиниці для перетворення з ортонормованим базисом), Δ_i (у відсотках) — відхилення L_2-норми i-ї базисної функції, яке в результаті проведеного аналізу становить Δ_i=0,15-0,42%, $i=\overline{0,7}$, неортогональність становить 0,07%.

3.3.2 Алгоритм швидкого обчислення 8-точкового прямого цілочислового модифікованого синус-косинусного перетворення типу VII

Матрицю C_4^* ядра ЦКП порядку 4 можна записати як добуток двох фактор-матриць:

$$C_4^* = C_2 C_1 , \tag{35}$$

де C_2, C_1 — фактор-матриці 4x4 алгоритму 4-точкового швидкого обчислення прямого ЦКП:

$$C_1 = H_4^*,\ C_2 = diag\left[T_2, Q_2\right],\ T_2 = \begin{bmatrix} g & g \\ g & -g \end{bmatrix},$$

$$Q_2 = \begin{bmatrix} f & e \\ -e & f \end{bmatrix},\ H_4^* = \begin{bmatrix} I_2 & \overline{I}_2 \\ \overline{I}_2 & -I_2 \end{bmatrix}, \tag{36}$$

H_4^* — фактор-матриця із ненульовими елементами ± 1.

Матрицю MSC_8^{VII*} згідно з (29) та з урахуванням алгоритму швидкого обчислення 4-точкового прямого ЦКП, а також алгоритму швидкого обчислення 4-точкового прямого ЦСП-VII можна факторизовано представити як добуток чотирьох матриць [16]:

$$MSC_8^{VII*} = T_{8,4} T_{8,3} T_{8,2} T_{8,1}, \tag{37}$$

де $T_{8,k}$ — k-і, $k = \overline{1,4}$, фактор-матриці 8x8 запропонованого алгоритму швидкого обчислення 8-точкового прямого цілочислового модифікованого синус-косинусного перетворення типу VII (ЦМСКП-VII):

$$T_{8,1} = H_8^*,\ T_{8,2} = diag\left[T_4, H_4^*\right],\ T_{8,3} = diag\left[R_4, T_2, Q_2\right],$$

$$T_{8,4} = diag\left[M_4, I_4\right],\ M_4 = \begin{bmatrix} M_{3,4} \\ \overline{S}_{1,4}^* \end{bmatrix}. \tag{38}$$

Тут M_4 — блочна матриця 4x4, при цьому її верхня частина представляє фактор-матрицю $M_{3,4}$, а нижня частина — вектор-рядок $S_{1,4}^*$.

3.3.3 Алгоритм швидкого обчислення 8-точкового оберненого цілочислового модифікованого синус-косинусного перетворення типу VII

Матрицю $\left(MSC_8^{VII}\right)^{-1}$ ядра оберненого ЦМСКП-VII порядку 8 можна отримати шляхом транспонування:

$$\left(MSC_8^{VII}\right)^{-1} = \left(SC_8^{VII*}\right)^T / k, \tag{39}$$

де $k = 2^m$, m — ціле.

Матрицю $\left(MSC_8^{VII}\right)^{-1}$ на основі (2.47), (2.45), (2.46) з урахуванням симетричності фактор-матриць ($H_8^{*T} = H_8^*$, $H_4^{*T} = H_4^*$) можна факторизовано представити як добуток чотирьох обернених фактор-матриць [16]:

$$\left(MSC_8^{VII}\right)^{-1} = T_{8,1} T_{8,2}^{-1} \overline{T}_{8,3}^T T_{8,4}^{-1}, \tag{40}$$

де $\overline{T}_{8,3}^T$, $T_{8,k}^{-1}$ — k-і, $k = 2,4$, відповідно транспонована і обернені фактор-матриці 8x8 запропонованого алгоритму швидкого обчислення 8-точкового оберненого ЦМСКП-VII:

$$\begin{gathered} T_{8,1} = H_8^*,\ T_{8,2}^{-1} = diag\left[M_{4i}^{-1}, H_4^*\right], \\ \overline{T}_{8,3} = diag\left[\overline{R}_4, \overline{T}_2, \overline{Q}_2^T\right],\ T_{8,4}^{-1} = diag\left[T_4^{-1}, I_4\right], \\ M_{4i}^{-1} = \left[\frac{M_{3,4}^{-1}}{\overline{S}_{1,4i}^{-1}}\right],\ \overline{S}_{1,4i}^{-1} = \left(-c, c, c, 0\right) / k. \end{gathered} \tag{41}$$

При цьому фактор-матриця $M_{3,4}^{-1} \neq M_{3,4}^T$ та містить у кожному рядку по три ненульові елементи ± 1, фактор-матриця $T_4^{-1} \neq T_4^T$ і містить у трьох рядках по два ненульові елементи ± 1,

$$\bar{R}_4 = R_4 / k = diag\left[d,c,a,b\right] / k\ ,\ \bar{T}_2 = T_2 / k = H_2,$$

$$\bar{Q}_2^T = Q_2^T / k\ ,\ \bar{Q}_2^T = \begin{bmatrix} f & -e \\ e & f \end{bmatrix} / k\ ,$$

де H_2 — матриця Адамара розміру 2x2, $H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$.

3.3.4 Обчислювальна складність цілочислового модифікованого синус-косинусного перетворення типу VII

Під час реалізації швидкого обчислення цілочислових перетворень використовуються операції «метелик» (butterfly), де виконуються парні множення, які можна реалізувати операціями лише зсуву і додавання, але в деяких випадках (для зменшення обчислювальної складності) — і за допомогою операції множення.

Для реалізації запропонованого алгоритму швидкого обчислення 1D 8-точкового однонормового ЦМСКП-VII-1 застосовано здебільшого операції зсуву і додавання, а також деякі операції множення. У табл. 20 [16] представлено схему обчислення спеціальних парних множень, які використовуються в операціях «метелик» для реалізації запропонованого алгоритму швидкого обчислення 1D 8-точкового оберненого ЦМСКП -VII-1.

Обчислювальну складність запропонованих алгоритмів швидкого обчислення 1D 8-точкових обернених ЦМСКП-VII-1 і відомих алгоритмів ДСП-VII [93] і ЦСП-VII [26,85] перетворень наведено у табл. 21 [16].

Розроблений швидкий алгоритм порівняно з відомим швидким алгоритмом [93] із складною структурою на сім ітерацій має просту регулярну структуру на чотири ітерації, що на три ітерації менше.

Таблиця 20

Множники операцій «метелик» матриці $\bar{Q}_2^T$ і елементів матриці $\bar{R}_4$		Алгоритм виконання операцій $y = f * x;$ $z = e * x$	Число операцій для реалізації «метелик»			Кількість використаних операцій
			Додавання	Зсув	Множення	
$f = 14/32$	$e = 43/32$	$x_1 = x - (x >> 3);$ $y = x_1 >> 1;\ x_2 = x + x_1;$ $z = x_1 + (x_2 >> 2)$	3	3	–	2
$a = 14/32$	0	$x_1 = x - (x >> 3);$ $y = x_1 >> 1;\ z = 0$	1	2	–	1
$b = 28/32$	0	$y = x - (x >> 3);\ z = 0$	1	1	–	1
$c = 37/32$	0	$x_1 = 37 * x;\ y = x_1 >> 5;$ $z = 0$	–	1	1	2
$d = 42/32$	0	$x_1 = 21 * x;\ y = x_1 >> 4;$ $z = 0$	–	1	1	1
Усього		–	8	12	3	–

Для реалізації алгоритму швидкого обчислення 1D 8-точкового оберненого ЦКП потрібно дві операції множення, 40 операцій додавання і 16 операцій зсуву.

Таблиця 21

Операції	Оцінка обчислювальної складності 1D 8-точкових обернених перетворень			Порівняльний аналіз ЦМСКП -VII-1 відносно	
	Запропонованого ЦМСКП-VII-1	Відомих ДСП-VII, ЦСП-VII			
	Повна факторизація	Матричне множення	Повна факторизація	[85]	[93]
	ЦМСКП -VII-1	ЦСП-VII із [85]	ДСП-VII із [93]		
Множення	3	64	21	У 21 раз менше	У 7 разів менше
Додавання+ зсув	35+12	56	77	На 16 % менше додавань	На 39% менше додавань

Таблиця 22

Операції (опер/ піксель)	Оцінка обчислювальної складності 2D роздільних адаптивних обернених перетворень для блоків 8х8			Порівняльний аналіз ЦКП+ ЦМСКП-VII-1 відносно	
	Запропонованого ЦКП+ ЦМСКП-VII-1	Відомих ЦКП+ДСП-VII (ЦКП+ЦСП-VII)		Н.265[90] + [85]	[51]+ [93]
		Н.265[90] + [85]	із [51]+ [93]		
Множення	0,625	10,75	4,125	У 17,2 раза менше	У 6,6 раза менше
Додавання+зсув	12,875	11,5	12,875	На 12 % більше додавань	0

Обчислювальну складність запропонованого алгоритму швидкого обчислення роздільних адаптивних ЦКП+ЦМСКП-VII-1, відомих ЦКП (Н.265) [90] + ЦСП-VII [85] і ЦКП [90]+ДСП-VII [93] 2D 8-точкових обернених цілочислових перетворень наведено у табл. 22 [16].

3.4 Висновки до розділу

Запропоновано цілочислове просте косинусне перетворення перетворення порядку 16 зі швидкими алгоритмами обчислення без виконання операцій множень, яке має у 2.2 рази менше операцій порівняно з відомим косинусним перетворенням порядку 16, і у 3-5 раз – меншу обчислювальну складність, з урахуванням складності операцій множення у порівнянні зі складністю операцій додавання і зсуву (в залежності від оцінки складності). Воно пропонується для використання в швидкодійному кодеку для збільшення швидкодії в якості альтернативи відомому косинусному перетворенню порядку 16, в першу чергу – при використанні алгоритму прогресивної передачі даних, оскільки при значній ефективності алгоритмів передбачення на основі середнього значення блоку

різниця ефективності перетворень є незначною. Для алгоритмів передбачення на основі сусідніх блоків розміром 16x16 ефективність косинусного перетворення порядку 16 на 3-5% вища за ефективність розробленого простого косинусного перетворення порядку 16.

На основі відомого методу побудовано цілочислове модифіковане синус-косинусне перетворення типу VII порядку 8 і розроблено алгоритм його швидкого обчислення, що має низьку мультиплікативну складність (в 7 раз менша) і потребує на 23,3% менше операцій додавання порівняно з відомим алгоритмом дискретного синусного перетворення типу VII.

Запропоновано цілочислове однонормове модифіковане синус-косинусне перетворення типу VII порядку 8, яке пропонується використовувати в рамках запропонованого адаптивного режиму роздільного адаптивного кодування з вибором з трьох ортогональних перетворень (косинусного, синус-косинусного і синусного) або в якості альтернативи синусному перетворенню для роздільного адаптивного кодування з вибором з двох перетворень (косинусного і синусного).

Запропонований алгоритм швидкого обчислення цілочислового модифікованого синус-косинусного перетворення типу VII порядку 8 має просту регулярну структуру на чотири ітерації, що на три ітерації менше порівняно з відомим алгоритмом із складною структурою на сім ітерацій, що додатково зменшує час обчислення.

Розроблено алгоритм швидкого обчислення 4-точкового оберненого цілочислового синусного перетворення типу VII для відомого цілочислового однонормового синусного перетворення типу VII порядку 4, запропонованого в роботі [15].

РОЗДІЛ 4
НОВІ ПІДХОДИ, МЕТОДИ І АЛГОРИТМИ АДАПТИВНОГО КОДУВАННЯ ЗОБРАЖЕНЬ

Для адаптивного кодування зображень та кадрів пропонуються нові підходи, методи і алгоритми, що базуються на розробленій математичній моделі кодування і стиснення зображень з втратами.

Запропоновано критерії вибору перетворень в залежності від статистичних характеристик вхідних даних, параметра квантування коефіцієнтів перетворень та швидкодії перетворень.

Для збільшення точності оцінки похибки запропоновано модифікацію міри MSE. Також запропоновано два універсальні алгоритми передбачення значень пікселів поточного блоку на основі зовнішніх даних (для прогресивної передачі даних) і на основі сусідніх блоків.

Для більш точної оцінки відстані в просторі кольорів розроблено нову векторну міру, яка враховує не лише відстань, подібно до евклідової мірі, але й зміни відтінків кольорів, пропорцій між основними кольорами. Розглянуто ідею застосування векторної міри для збільшення точності обчислення похибки відновлення зображень. Новий поштовх для подальшого розвитку отримав відомий підхід виділення індексованих палітр кольорів.

4.1 Адаптивне використання перетворень

Адаптивне використання перетворень є важливою складовою підвищення ефективності стиснення. Першим алгоритмом було адаптивне використання косинусних перетворень порядку 4 і 8 в H.264 [6]. В кодеку наступного покоління H.265 крім адаптивного вибору з косинусних перетворень порядків 4, 8, 16 і 32 були застосовані роздільні адаптивні перетворення порядку 4, що використовували косинусне і синусне перетворення порядку 4 [26]. В наступному кодеку, H.266, кількість перетворень для адаптивного підбору

значно виросла. Але всі існуючі алгоритми базуються на емпіричному підборі оптимальних перетворень, не використовуючи попередніх оцінок для зменшення обчислювальної складності процесу порівняння й вибору. Запропоновано ряд критеріїв для спрощеного попереднього вибору перетворень, що особливо актуально для швидкодійних кодеків.

4.1.1 Вибір перетворень в залежності від кореляції вхідних даних

Залежність між статистичними характеристиками вхідних даних і оптимальним вибором порядку косинусного перетворення давно відома: чим вища кореляція між даними, тим вищого порядку перетворення доцільно використовувати. Але прості алгоритми для оцінки статистичних характеристик зображень/кадрів з метою попереднього вибору найбільш відповідного перетворення до всього зображення на практиці не використовуються, оскільки емпіричний підбір перетворень є більш точним.

В той же час такий підхід дозволяє отримати більшу ефективність ніж фіксований розмір перетворення, і є проміжним по ефективності і обчислювальній складності між фіксованим і локально адаптивним підходами.

В залежності від статистичних характеристик вхідних даних можна обирати як одне перетворення для всього зображення, так і два перетворення для подальшого адаптивного порівняння і вибору, як це було показано вище.

4.1.2 Вибір перетворень в залежності від коефіцієнта стиснення

Коефіцієнт стиснення залежить від параметру квантування коефіцієнтів перетворення і від статистичних характеристик вхідних даних. Відповідно, в залежності від цих двох параметрів можна попередньо обирати порядок перетворення для всього зображення.

При великих коефіцієнтах стиснення ефективніше застосовувати перетворення більшого порядку, 16 і інколи 32, оскільки чим більше порядок

перетворення, тим краще воно зберігає низькочастотні складові зображення, хоча й втрачає високочастотні. Але при великих параметрах квантування високочастотні складові втрачаються дуже сильно, що й обумовлює вибір. В свою чергу для малих і середніх коефіцієнтів стиснення краще підходить перетворення порядку 8, що краще зберігає високі частоти.

У випадку використання універсального алгоритму передбачення значень пікселів на основі прогресивної передачі даних ефективність кодування перетворень значно зменшується, як показано вище. Відповідно, порядок перетворення відіграє меншу роль, ніж при традиційних підходах, а обчислювальна складність починає грати більшу вагу.

4.1.3 Вибір перетворень за умови мінімізації обчислень

Для швидкодійних кодеків, які мають пріорітет на зменшенні обчислювальної складності кодування, додатковою умовою вибору перетворення є необхідність мінімізації обчислень.

Ця умова має більшу вагу у виборі перетворення при великих коефіцієнтах стиснення, тобто великому параметрі квантування коефіцієнтів перетворення і при високій кореляції початкових даних.

При такій комбінації умов доцільним є застосування перетворень з меншою обчислювальною складністю, наприклад, ступінчастих перетворень Уолша-Хаара, які мають швидкі алгоритми без виконання операцій множення і більш ніж в два рази меншу загальну кількість операцій на піксель відносно простих косинусних перетворень.

Проміжними по якості і швидкодії між ступінчастими перетвореннями і косинусними перетвореннями є прості косинусні перетворення. Відповідно, прості косинусні перетворення часто є оптимальними, особливо у випадку наявності швидких алгоритмів без виконання операцій множення.

4.2 Модифікація оцінок MSE і PSNR

Запропоновано наступну модифікацію обчислення похибок – похибка обчислюється лише для тих пікселей, для яких похибка більше певного порогового значення. Це дає більш точну картину розподілу похибок, а також, в якості додаткового ефекту – зменшення обчислювальної складності обчислення MSE і PSNR, оскільки для обчислення враховується менша частина пікселей.

Підставою для виникнення ідеї модифікації обчислення похибки виступив той факт, що людське око в першу чергу враховує помітні відхилення, більші за певне значення, і ігнорує малі похибки. Отже, є доцільним ігнорувати малі відхилення від оригіналу, і враховувати значення похибок лише більших за певне X, обраного порогового значення.

При такому підході очікуваним виявилось загальне збільшення середнього значення похибок, оскільки значні відхилення не розсіювались на фоні великої кількості малих відхилень.

Неочікуваним виявилось значно більші значення похибок для косинусних перетворень порядку 32 відносно інших порядків, 8 і 16. В загальній статистиці тестування усереднена по всьому зображення різниця між ними незначна, і перетворення порядку 32 розглядались як найбільш ефективні за коефіцієнтом стиснення (хоча статистика частоти адаптивного вибору перетворень різних порядків також свідчить, що найчастіше використовуються перетворення порядку 8 і 16, а порядків 32 і 4 – значно менше). Таким чином, концепція запропонованої міри краще пояснює статистику, ніж існуючі теорії.

4.3 Універсальні алгоритми передбачення значень пікселів

Універсальні, тобто фіксовані, алгоритми передбачення значень пікселів поточного блоку зображення/кадру характеризуються низькою складністю, оскільки відсутня необхідність адаптивного вибору найкращого алгоритму з множини можливих. Це має призводити до відповідного зниження ефективності

стиснення, але згідно з результатами тестових порівнянь кодеку LCEVC [52] для більшості статичних сцен різниця незначна.

Універсальний алгоритм передбачення вперше був реалізований в рамках прогресивної передачі даних в кодеку LCEVC, при цьому значення пікселя зображення/кадру меншого масштабу використовується в якості передбачення для блоку 2x2 або 4x4 більшого масштабу.

Використання алгоритму представлення даних у формі зображення меншого масштабу і різниць декількох рівнів дозволяє спростити загальну обчислювальну складність кодування, використовуючи універсальний алгоритм передбачення значень пікселів.

4.3.1 Універсальний алгоритм передбачення на зовнішніх даних

Пропонується новий алгоритм формування зображень меншого масштабу для використання в рамках прогресивної передачі даних. В рамках запропонованого підходу обирається таке значення пікселя зображення/кадру меншого масштабу, яке дає меншу похибку, з двох можливих варіантів значень – середнього значення та медіанного значення блоку більшого масштабу.

Відповідно, алгоритм передбачення використовує значення пікселя меншого масштабу в якості передбачення значень всіх пікселів блоку більшого масштабу.

Обчислювальна складність алгоритму передбачення на зовнішніх даних є меншою, оскільки не потребує обчислення відновлених даних, тобто оберненого перетворення, як це необхідно для алгоритмів передбачення на основі сусідніх блоків зображення.

4.3.2 Універсальний алгоритм інтра-передбачення

Для швидкодійного кодека пропонується використовувати фіксований, тобто універсальний алгоритм передбачення значень пікселів поточного блоку

на основі значення одного пікселя сусіднього блоку. Це збільшує коефіцієнт стиснення в середньому на 30%, хоча й супроводжується незначним зниженням PSNR, як було показано вище, до 0,5 дБ, порівняно з варіантом кодування без передбачення. В якості сусіднього блоку береться попередній блок зліва, а передбачення робиться на основі сусіднього пікселя рядом з першим пікселем поточного блоку (на рис. має позначення «I»).

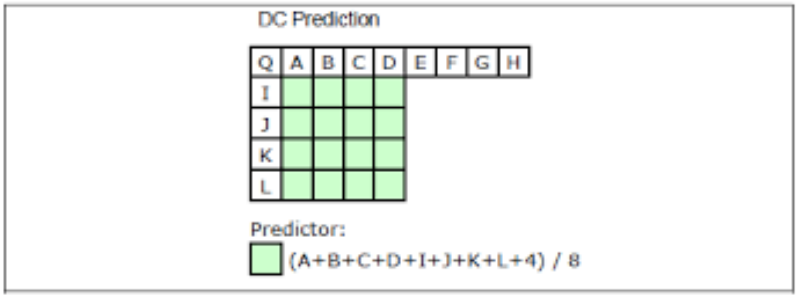

Рисунок 16. DC-передбачення

Для оцінки ефективності стиснення було зроблено порівняння з відомим алгоритмом DC-передбачення по середньому на основі середнього значення найближчого стовпця блоку зліва і найближчого рядка блоку зверху.

Результати тестування на трьох тестових зображеннях виявились практично однаковими, іноді відрізняючись на 0,01 дБ., тому вони не наводяться. Таким чином, запропонований алгоритм може бути використаний в якості універсального, тобто без адаптивного вибору найкращого з декількох можливих, з метою мінімізації обчислювальної складності.

4.4 Векторна міра обчислення відстані простору кольорів

Найбільш поширений алгоритм обчислення відстані в просторі кольорів є використання евклідової міри. Але евклідова метрика використовується для просторів, які не мають додаткових інтегральних параметрів. Адже колір

визначається трьома складовими, що принципово відрізняється від інших просторів, де застосовується евклідова відстань.

Тобто евклідова метрика не буде розрізняти два наступні варіанти відхилень від початкового кольору:

1) варіант збільшення (+x, +x, +x), варіант зменшення (-x, -x, -x) (тобто одночасне збільшення або зменшення по всім каналам кольорів RGB);
2) зміни відтінків кольорів: (+, -, +) або (-, +, +) або (+, +, -), коли два значення змінюються в одному напрямку, а третє значення – в іншому напрямку, незалежно від позиції. Хоча відтінки звісно будуть різні, в кожному випадку.

При цьому людське око буде розрізняти перший варіант і другий, оскільки в першому випадку змінюється яскравість загального кольору, а не відтінок, а в другому випадку – змінюється саме відтінок кольорів (вплив/вага одного з складових кольорів, співвідношення складових кольорів в загальному кольорі).

Таким чином, пропонується модифікувати алгоритм обчислення відстані в просторі кольорів з врахуванням змін яскравості та відтінків, оцінюючи зміни співвідношення значень пікселів окремих каналів кольору RGB:

$$V = \sqrt{\left(\frac{r_1}{g_1} - \frac{r_2}{g_2}\right)^2 + \left(\frac{r_1}{b_1} - \frac{r_2}{b_2}\right)^2 + \left(\frac{g_1}{b_1} - \frac{g_2}{b_2}\right)^2}$$

де *r*, *g* та *b* – значення пікселів за кольорами простору RGB.

Обчислення векторної відстані в просторі кольорів може використовуватись як для побудови індексованих палітр кольорів (тобто обмеженої кількості кольорів, що індексуються, тобто нумеруються, і зберігаються для кожного зображення, як в форматі GIF), так і для обчислення похибки відновлення.

4.4.1 Векторна міра для побудови індексованих палітр кольорів

Індексовані палітри будуються шляхом знаходження і нумерації (наприклад, з 1 до 256) обмеженої кількості кольорів з найбільшим ваговим

коефіцієнтом для конкретного зображення, всі інші кольори замінюються на найближчий колір зі отриманої індексованої палітри. Іншими словами, кожному номеру з 1 до 256 відповідає певний колір, тобто набір значень RGB. І повний простір RGB для конкретного зображення замінюється на спрощений простір, тобто на індексовану палітру.

Зрозуміло, що такий підхід найкраще працює для зображень з незначною кількістю основних кольорів, наприклад, для інфографіки. І навпаки, для зображень з великою кількістю різних кольорів він підходить найгірше.

Представляє цікавість проміжний варіант – зображення з середньою кількістю основних кольорів, що потребують збільшеної палітри (відносно 8 бітної). Але збереження наприклад 10 бітної палітри (яка має 1024 кольори) вже потребує в 4 рази більше службової інформації. Виникає цікава і актуальна задача побудови простору індексованих палітр та запису конкретної індексованої палітри шляхом певних параметрів та/або відхилень від типових значень палітр, що входять в цей простір, або побудови гібридних індексованих палітр, що використовують спрощені простори кольорів зі зменшеною кількістю біт на піксель на кожну складову кольору.

Зазвичай для побудови індексованої палітри (квантування кольорів) використовуються декілька алгоритмів. На даний момент існує кілька методик квантування кольору. Одним з найбільш відомих є метод медіанного перетину [94]. При цьому колірний простір розглядається як тривимірний куб. Кожна вісь куба відповідає одному з трьох основних кольорів: червоного, зеленого або синього. Кожна з трьох сторін розбивається на 255 рівних частин, поділу на осях нумеруються від 0 до 255, причому більше значення відповідає більшій інтенсивності кольору. Метод медіанного перетину ділить куб на 256 паралелепіпедів, кожен з яких містить приблизно однакову кількість пікселів. При такій розбивці куба центральна точка кожного паралелепіпеда представляє оптимальний вибір для кольорової палітри. У тій області куба, яка густо заповнена точками, буде більше паралелепіпедів і, відповідно, в палітру потрапить більше кольорів. А там, де точок менше, буде взято меншу кількість

кольорів. При цьому жоден колір не буде відкинутий повністю, а перевага буде віддана тим кольорам, які зустрічаються частіше. Припустимо, у нас є зображення з довільним числом пікселів та кольорів і потрібно створити палітру з 16 кольорів. Необхідно зібрати всі пікселі зображення (тобто їх значення RGB) у список, і визначити, який колірний канал (червоний, зелений або синій) серед пікселів в списку має найбільший діапазон, а потім сортувати пікселі відповідно до значень цього каналу. Наприклад, якщо синій канал має найбільший діапазон, піксель з значенням RGB (32, 8, 16) буде меншим, ніж піксель зі значенням RGB (1, 2, 24), оскільки B(16) <B(24). Після відсортування потрібно скопіювати верхню половину пікселів у новий список. (Саме цей крок дає алгоритму медіанного перетину його назву, списки розділяються на дві частини посередині у списку пікселів.) Доведеться повторити процес на обох списках, отримавши 4 списка, потім повторити ітерацію декілька разів. Оскільки кількість списків подвоюється з кожною ітерацією, цей алгоритм може генерувати лише палітру з кількістю кольорів, що є степенем двійки.

Другий алгоритм квантування кольорів – застосування октодерева. Він простіший, але менш точний, тому не представляє цікавості.

Третій алгоритм побудови індексованої палітри - алгоритм кластеризації простору кольорів, виділення ієрархічної структури кластерів/підгруп кольорів. Алгоритм може бути представлений наступними кроками:

0. За N шуканих кластерів приймаються будь-які N точок множини, наприклад перші за номером.

1. Для кожного знайденого кластера розраховуємо центр – середнє значення (центр мас) всіх об'єктів, включених в нього.

2. Для кожного об'єкта розраховуємо набір відстаней до кожного з N кластерів.

3. Кожен об'єкт відносимо в той кластер, відстань до якого мінімальна.

4. Переходимо до кроку 1, до тих пір, поки не задоволений певний формальний критерій якості кластеризації. Наприклад, ітерації зупиняються,

якщо кількість об'єктів, які змінили свій кластер на поточному кроці, відповідає кількості таких же об'єктів на попередньому кроці [94].

Відповідно, для кластерізації кольорів потрібно використовувати оцінку відстані між кольорами. Для цього зазвичай використовують евклідову міру, іноді – зважену евклідову міру, бо наприклад формула кольорової відмінності CIEDE2000 занадто складна для багаторазового обчислення, тому вона рідко використовується.

В якості кращої альтернативи зваженій евклідовій метриці пропонується векторна міра простору кольорів.

4.4.2 Векторна міра для оцінки похибки відновлення

Іншим варіантом застосування векторної міри є оцінка похибки відновлення зображення/кадру, тобто порівняння і оцінка різниці між оригінальним і відновленим, декодованим зображенням.

Кодування на основі ортогональних перетворень, як відомо, має свої недоліки, наприклад, взаємовплив даних, що кодуються, що призводить до спотворення оригінальних чистих кольорів, розмиття контурів, зменшення або збільшення значень під впливом сусідніх пікселів. Похибки залежать від степені стиску, кореляції даних, використаних перетворень та інш.

Причому векторна міра є доповненням евклідової міри, тобто можна побудувати комплексну інтегральну міру K, яка включає в себе дві складові, евклідову E і векторну B міри:

$$K=E+B\text{i},$$

яка є аналогічною до запису комплексного числа. Тобто евклідова міра є аналогом дійсної складової, а векторна міра – аналогом уявної складової, яка активується певним чином, в даному випадку – коли виконується умова різнонаправлених змін.

Тобто *E*, евклідова міра, обчислюється для обох альтернативних варіантів – і коли зміни однонаправлені (одночасне збільшення або зменшення для всіх трьох шкал кольорів), і коли зміни різнонаправлені. А друга складова (векторна міра) в першому варіанті дорівнює нулю, і лише в другому варіанті доповнює, уточнює евклідову міру.

4.5 Індексовані палітри та адаптивний вибір простору кольорів

Індексовані палітри, як звичайні, так і поглиблені (на 512 або 1024 кольори) можуть адаптивно застосовуватись разом з повнокольоровими просторами типу RGB або різних режимів YCrCb.

Алгоритм оцінки та вибору одного з варіантів досить простий: достатньо обчислити кількість кольорів зображення з певною обраною похибкою P між основними кольорами і допоміжними, додатковими, і якщо кількість кольорів буде менше певного обраного M, то застосовується індексована палітра. Якщо M достатньо мале, цей критерій буде спрацьовувати лише для зображень з малою кількістю кольорів, що найчастіше є інфографікою. Якщо ж M відносно велике, то це буде варіант середньої кількості кольорів, який потребує поглибленої індексованої палітри або спрощеного простору кольорів (наприклад, 16 біт замість стандартного 24 бітного кольору).

Аналогічно, можна запропонувати інший підхід: адаптивного вибору між просторами кольорів RGB і різними режимами YCrCb. Хоча YCrCb в режимі 4:4:4 має на 7% менший об’єм даних, ніж в RGB, але це вартує приблизно 1 дБ. втрати якості за характеристикою PSNR. Інші режими мають ще більший стиск, і більшу похибку. Відповідно, для випадків найкращої якості обирається простір кольорів RGB, в інших випадках – різні режими простору YCrCb. Хоча з урахуванням постійного зростання якості відновлених зображень, простір YCrCb може втратити свою поширеність. З іншого боку, як вже було зазначено, застосування нейромереж для наближеного відновлення кольорів є перспективним підходом, адже в такому випадку

достатньо використовувати лише спрощені простори кольорів чи навіть індексовані палітри, що дозволить значно збільшити теоретично можливі коефіцієнти стиснення.

Підхід використання індексованих палітр кольорів є значно обмеженим, і навіть поглиблені палітри кольорів не зроблять його універсальним. Але подальший розвиток алгоритмів стиснення зображень/кадрів з втратами може бути саме в цьому напрямку, що дозволить помітно збільшити коефіцієнт стиснення, аналогічно використанню алгоритмів передбачення (що в середньому на 20-30% збільшує стиснення, при незначному зменшенні якості). Отже, виникає актуальна задача побудови індексованих палітр динамічної розмірності (тобто адаптивно до конкретного зображення/кадру) з мінімальним об’ємом відповідної службової інформації.

Другий напрямок досліджень в просторі кольорів – використання гібридного підходу, тобто узгодженого застосування індексованих палітр і просторів кольорів. Можливо, одним із варіантів для гібридних реалізацій буде використано новий простір кольорів, що побудований на пропорціях між кольорами.

4.6 Висновки до розділу

Для адаптивного кодування зображень та кадрів пропонуються нові підходи, методи і алгоритми, що базуються на розробленій математичній моделі кодування і стиснення зображень з втратами.

Запропоновано критерії вибору перетворень в залежності від статистичних характеристик вхідних даних, параметра квантування коефіцієнтів перетворень та швидкодії перетворень.

Запропоновано два універсальні алгоритми передбачення значень пікселів поточного блоку на основі зовнішніх даних (для прогресивної передачі даних) і на основі сусідніх блоків.

Для збільшення точності оцінки похибки запропоновано модифікацію міри MSE. Це дає більш точну картину розподілу похибок, а також, в якості додаткового ефекту – зменшення обчислювальної складності обчислення MSE і PSNR, оскільки для обчислення враховується менша частина пікселей.

Для більш точної оцінки відстані в просторі кольорів розроблено нову векторну міру, яка враховує не лише відстань, подібно до евклідової мірі, але й зміни відтінків кольорів, пропорцій між основними кольорами.

Обчислення векторної відстані в просторі кольорів може використовуватись для побудови індексованих палітр кольорів, для обчислення похибки відновлення і для створення нових просторів кольорів на основі співвідношення між кольорами RGB.

ВИСНОВКИ

На основі запропонованої математичної моделі висунуто ідею оптимальних кодеків для кожного рівня обчислювальної потужності, дослідження якої привело до нової концепції адаптивних кодеків різного рівня складності – швидкодійних, збалансованих, ресурсоємних і універсальних.

В рамках концепції адаптивних кодеків різного рівня запропоновано додавання адаптивних режимів низької складності, що використовуються для швидкодійного кодека, в кодеки вищих рівнів в якості складової частини.

Необхідність вибору і переключення між адаптивними режимами різної складності стало підгрунтям для введення додаткового ієрархічного рівня в структурі кодека, що породжує нову архітектуру кодування.

Для швидкодійного кодека в якості доповнення, альтернативі режимам низької складності додатково запропоновано так званий екстремальний режим низької складності, який має ще менші мінімальні вимоги до обчислювальних потужностей.

Використання в кодеках вищих рівнів складності адаптивних режимів, що використовуються в кодеках попередніх рівнів складності, породжує можливість зворотньої сумісності кодеків.

В рамках розвитку концепції адаптивних кодеків різного рівня для кожного класу кодеків запропоновано ряд нових алгоритмів.

Для швидкодійного кодека запропоновано наступні алгоритми, модифікації та покращення у порівнянні з кодеком H.263:

- універсальний алгоритм передбачення значень пікселів;
- цілочислові косинусні перетворення зі швидкими алгоритмами;
- просте цілочислове косинусне перетворення порядку 16 зі швидкими алгоритмами без операцій множення для окремих режимів кодування;
- алгоритм вибору перетворень в залежності від коефіцієнта стиснення;
- простий алгоритм адаптивного вибору перетворень порядку 8 і 16 для всього зображення/кадру;

- екстремальні режими кодування наднизької обчислювальної складності;
- використання спрощених просторів кольорів.

В результаті отримаємо швидкодійний кодек, що потребує менше обчислювальних ресурсів ніж оригінальний кодек H.263, але з коефіцієнтом стиснення як у H.264.

Для збалансованого кодека (оптимального гібриду двох неоптимальних кодеків – H.264 і H.265) запропоновано наступні алгоритми, модифікації і покращення відносно кодеків H.264 і H.265:

- нові адаптивні режими середньої і високої обчислювальної складності;
- алгоритми вибору адаптивних режимів;
- адаптивні режими низької обчислювальної складності;
- зворотня сумісність з швидкодійним кодеком.

Для ресурсоємного кодека запропоновано наступні алгоритми, модифікації і покращення відносно кодеків H.265 і H.266:

- новий алгоритм адаптивного використання трьох типів перетворень: косинусного, синусного і синус-косинусного;
- нові адаптивні режими високої складності;
- цілочислове однонормове модифіковане синус-косинусне перетворення порядку 8 зі швидкими алгоритмами низької обчислювальної складності;
- використання адаптивних режимів низької і середньої складності кодеків попереднього рівня;
- зворотня сумісність зі збалансованим кодеком.

Запропоновано об’єднання всі трьох типів кодеків в універсальний кодек. Універсальний кодек характеризується більшою складністю алгоритмів вибору адаптивних режимів, у порівнянні зі спеціалізованими кодеками.

На основі розробленої мат. моделі запропоновано нові алгоритми вибору типу й порядку перетворення в залежності від статистичних характеристик вхідних даних і від коефіцієнту стиснення (параметру квантування).

Обгрунтовано використання прогресивної передачі даних з універсальним алгоритмом передбачення на основі зовнішніх даних (зменшеного зображення).

Запропоновано простий алгоритм адаптивного вибору порядку перетворення в залежності від статистичних характеристик конкретного зображення/кадру, що дає збільшення ефективності стиснення на 10-12% у порівнянні з фіксованим вибором перетворення порядку 8. Для зображень зі змінною кореляцією запропоновано адаптивний локальний вибір порядку перетворення за спрощеним алгоритмом.

Запропоновано підхід використання спрощених просторів кольорів та значних коефіцієнтів стиснення з метою подальшої обробки та покращення якості зображень/кадрів з застосуванням ШІ.

Запропоновано цілочислове просте косинусне перетворення перетворення порядку 16 зі швидкими алгоритмами обчислення без виконання операцій множень, яке має у 2.2 рази менше операцій порівняно з відомим косинусним перетворенням порядку 16, і у 3-5 раз – меншу обчислювальну складність, з урахуванням складності операцій множення у порівнянні зі складністю операцій додавання і зсуву (в залежності від оцінки складності). Воно пропонується для використання в швидкодійному кодеку для збільшення швидкодії в якості альтернативи відомому косинусному перетворенню порядку 16, в першу чергу – при використанні алгоритму прогресивної передачі даних, оскільки при значній ефективності алгоритмів передбачення на основі середнього значення блоку різниця ефективності перетворень є незначною. Для алгоритмів передбачення на основі сусідніх блоків розміром 16x16 ефективність косинусного перетворення порядку 16 на 3-5% вища за ефективність розробленого простого косинусного перетворення порядку 16.

На основі відомого методу побудовано цілочислове однонормове модифіковане синус-косинусне перетворення типу VII порядку 8 і розроблено алгоритм його швидкого обчислення, що має низьку мультиплікативну складність (в 7 раз менша) і потребує на 23,3% менше операцій додавання порівняно з відомим алгоритмом дискретного синусного перетворення типу VII.

Запропоноване цілочислове однонормове модифіковане синус-косинусне перетворення типу VII порядку 8 пропонується використовувати в рамках запропонованого адаптивного режиму роздільного адаптивного кодування з вибором з трьох ортогональних перетворень (косинусного, синус-косинусного і синусного) або в якості альтернативи синусному перетворенню для роздільного адаптивного кодування з вибором з двох перетворень (косинусного і синусного).

Запропонований алгоритм швидкого обчислення цілочислового модифікованого синус-косинусного перетворення типу VII порядку 8 має просту регулярну структуру на чотири ітерації, що на три ітерації менше порівняно з відомим алгоритмом із складною структурою на сім ітерацій, що додатково зменшує час обчислення.

Розроблено алгоритм швидкого обчислення 4-точкового оберненого цілочислового синусного перетворення типу VII для відомого цілочислового однонормового синусного перетворення типу VII порядку 4, запропонованого в роботі [15].

Запропоновано два універсальних алгоритми передбачення значень пікселів на основі зовнішніх (для прогресивної передачі) і внутрішніх даних (з метою зменшення обчислювальної складності).

Вперше запропоновано модифікація MSE, яка враховує лише значення, більші за обране порогове X. На основі модифікованої MSE обчислюється модифіковане PSNR, яке має підвищену точність відображення спотворень, особливо контурів та перепадів значень, а також значно зменшену обчислювальну складність.Така ж модифікація розглянута і для інших оцінок похибок.

Вперше для простору кольорів розроблена нова векторна міра, що враховує як зміни яскравості, так і зміни кольорів. Запропонована міра може використовуватись як для оцінки похибок кодування, так і для оцінки відстані між кольорами, зокрема при побудові локальних динамічних просторів кольорів.

Запропоновано адаптивні алгоритми вибору простору кольорів: індексованої палітри, RGB та різних режимів YcrCb.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Tung Nguyen, Detlev Marpe, "Compression efficiency analysis of AV1, VVC, and HEVC for random access applications", APSIPA Transactions on Signal and Information Processing, vol.10, no.1, 2021
2. Georgios Minopoulos, Vasileios A. Memos, Konstantinos E. Psannis, Yutaka Ishibashi, "Comparison of Video Codecs Performance for Real-Time Transmission", 2020 2nd International Conference on Computer Communication and the Internet (ICCCI), pp.110-114, 2020
3. Laude T, Adhisantoso YG, Voges J, Munderloh M, Ostermann J. A Comprehensive Video Codec Comparison. APSIPA Transactions on Signal and Information Processing. 2019;8:e30. doi:10.1017/ATSIP.2019.23
4. Salomon, David. "Data Compression: The Complete Reference". 4th Edition, Springer, 2007. 1091 p
5. Gonzalez, Rafael C., and Woods, Richard E. "Digital Image Processing". 3rd Edition, Prentice Hall, 2008. 954 p
6. Richardson, Iain E. G. "Video Coding: H.264 and MPEG-4 – Standards of the Next Generation". Academic Press, 2003. 302 p
7. Луц В.К. Нові підходи та методи адаптивного кодування зображень. Кібернетика і системний аналіз. 2024. Т. 60, № 2. С. 188-200. https://doi.org/10.34229/KCA2522-9664.24.2.16
8. Luts Ya.V., Luts V.K. About the Development of a High-Speed Simplified Image Codec. Cybernetics and Computer Technologies. 2021. 1. P. 61–66. (in Ukrainian) https://doi.org/10.34229/2707-451X.21.1.6
9. Луц В.К. Нові адаптивні кодеки, режими та алгоритми кодування зображень. 1-а Міжнародна науково-практична конференція «Інформаційні системи та технології: результати і перспективи», м. Київ, 6 березня 2024 р., с. 278-282. URI:https://www.ist.fit.knu.ua/news/abstracts-ist2024
10. Луц В. Комплексні підходи та моделі для кодування та обробки зображень. Міжн. наук.-пр. конф. «Інформаційні технології та комп’ютерне

моделювання», ІТКМ-2022, 15 – 16 грудня 2022 р., м. Івано-Франківськ, с. 60-62. https://itcm.comp-sc.if.ua/2023/ITCM-22_Zbirnyk_final.pdf

11. Луц В.К., Луц Я.В. «Адаптивні режими кодеків зображень», X Міжнародна науково-практична конференція Глушковські читання", м. Київ, 2 грудня 2021. С. 113-117 URI: http://glushkov.kpi.ua/GC21
12. Луц В.К. Перспективи застосування алгоритмів кластерізації для зменшення розмірності і складності математичних моделей. X Міжнародна науково-практична конференція: "Глушковські читання", м. Київ, 2 грудня 2021 р., с. 109-113. URI: http://glushkov.kpi.ua/GC21
13. Луц В.К. Адаптивний ітераційний метод для зменшення розрядності зображень і відеоданих/ Проблемно-наукова міжгалузева конференція «Юриспруденція та проблеми інформаційного суспільства» (ЮПІС – 2018), 23-25 травня 2018 р., м. Надвірна. С. 63-68.
14. Shuai Huo, Dong Liu, Li Li, Siwei Ma, Feng Wu, and Wen Gao Towards Hybrid-Optimization Video Coding. Image and Video Processing. 2022. https://doi.org/10.48550/arXiv.2207.05565
15. Hnativ, L.O., Luts, V.K. Algorithms for Fast Implementation of 4-Point Integer Sine Type VII Transforms without Multiplication and Separable Directional Adaptive Transforms for Intra Prediction in Image/Video Coding. Cybern Syst Anal 56, 159–170 (2020). https://doi.org/10.1007/s10559-020-00231-y
16. Hnativ, L.O., Luts, V.K. Integer Modified Sine-Cosine Transforms Type VII. A construction Method and Separable Directional Adaptive Transforms for Intra Prediction with 8 × 8 Chroma Blocks in Image/Video Coding. Cybern Syst Anal 57, 155–164 (2021). https://doi.org/10.1007/s10559-021-00339-9
17. Луц В., Луц Я. Про використання модифікованого дельта-алгоритму для обробки та кодування зображень. Фізико-математичне моделювання та інформаційні технології, (36), 148-152. (2023). https://doi.org/10.15407/10.15407/fmmit2023.36.148
18. Мюрей Д.Д., Райпер У. Енциклопедія форматів графічних файлів. BHV-

Київ, 1997. — 672 с

19. ISO/IEC 23008-12:2017 – Information technology – Hight efficiency coding and media delivery in heterogeneous environments – Part 12: Image File Format. URL: https://www.iso.org/standard/66067.html
20. HEIF – High Efficiency Image File Format. URL: https://nokiatech.github.io/heif/ Дата звернення 11.10.2023
21. BPG Image Format. URL: https://bellard.org/bpg/ Дата звернення 11.10.2023
22. AV1 Image File Format (AVIF) URL: https://aomediacodec.github.io/av1-avif/ Дата звернення 11.10.2023
23. A new image format for the Web | WebP | Google Developers. URL: https://developers.google.com/speed/webp/ Дата звернення 11.10.2023
24. FLIF – Free Lossless Image Format. URL: http://flif.info/ Дата звернення 11.10.2023
25. JPEG. URL: https://jpeg.org Дата звернення 11.10.2023
26. ITU-T Rec. H.265/ISO/IEC 23008-2: 2013. Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 2: High efficiency Video Coding, 2013
27. G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard", IEEE Trans. Circuits and Systems for Video Technology, Vol. 22, No. 12, pp. 1649-1668, Dec. 2012.
28. J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the Coding Efficiency of Video Coding Standards - Including High Efficiency Video Coding (HEVC)", IEEE Trans. Circuits and Systems for Video Technology, Vol. 22, No. 12, pp. 1669-1684, Dec. 2012.
29. F. Bossen, B. Bross, K. Sühring, and D. Flynn, "HEVC Complexity and Implementation Analysis," IEEE Trans. Circuits and Systems for Video Technology, Vol. 22, No. 12, December 2012.
30. Versatile Video Coding (VVC) | JVET. URL: https://jvet.hhi.fraunhofer.de/ Дата звернення 11.10.2023
31. L.O. Hnativ. New integer fractal step multi wavelets and their fast transform for

high efficiency image and video coding // Issues Optimization Computing (IOC-XLII): Proceedings Intern. Conf., Zakarpattya region 21 – 25 September 2015.

32. L.O. Hnativ, V.K. Luts. High efficiency image and video coding based on integer wavelet transforms at the basis of the fractal piecewise linear step functions // Issues Optimization Computing (IOC-XLII): Proceedings Intern. Conf., Zakarpattya region 21 – 25 September 2015
33. L.O. Hnativ. Fractal step multiwavelets - a new wavelet technology for signal processing and image encoding. Theses of International Scientific Conference "Modern Informatics: Problems, Achievements and Prospects for Development", dedicated to the 60th anniversary of the V.M. Glushkov Institute of Cybernetics of NAS of Ukraine. December 13-15, 2017. Kyiv, V.M. Glushkov Institute of Cybernetics of NAS of Ukraine, 2017. pp 195-197
34. L.O. Hnativ, Luts V. K. New image and video encoding technology based on integer fractal stepped multiwavelets. Theses of International Scientific Conference "Modern Informatics: Problems, Achievements and Prospects for Development", dedicated to the 60th anniversary of the V.M. Glushkov Institute of Cybernetics of NAS of Ukraine. December 13-15, 2017. Kyiv, V.M. Glushkov Institute of Cybernetics of NAS of Ukraine, 2017. pp 197-199.
35. Hnativ, L. (2021). Ортонормовані базиси фрактальних ступінчастих мультивейвлетів – нова мультивейвлет-технологія для оброблення сигналів і зображень. Фізико-математичне моделювання та інформаційні технології, (32), 91-95. https://doi.org/10.15407/fmmit2021.32.091
36. Hnativ, L. (2023). Дискретні фрактальні ступінчасті мультивейвлети і мультивейвлет-пакети – нова мультивейвлет-технологія для оброблення та кодування зображень. Фізико-математичне моделювання та інформаційні технології, (36), 53-57. https://doi.org/10.15407/10.15407/fmmit2023.36.053
37. Lev Hnativ, Vasyl Luts, Yaroslav Luts. Fractal Step Multiwavelets and Multiwavelet Packets - A New Multiwavelet Technology for Image Processing and Coding. III International Scientific Symposium “Intelligent Solutions” (IntSol-2023). Symposium Proceedings Kyiv - Uzhhorod, Ukraine, September

27-28, 2023, p. 282-294

38. Луц В.К. Просте цілочисельне косинусне перетворення порядку 16 для кодування сигналів і зображень. Штучний інтелект. 2014. № 4. С. 31–43

39. Hnativ, L.O. Integer Cosine Transforms: Methods to Construct new Order 8, 16 Fast Transforms and Their Application. Cybern Syst Anal 50, 913–929 (2014). https://doi.org/10.1007/s10559-014-9682-9

40. Hnativ, L.O. Integer Cosine Transforms for High-Efficiency Image and Video Coding. Cybern Syst Anal 52, 802–816 (2016). https://doi.org/10.1007/s10559-016-9881-7

41. Hnativ, L.O. A Method to Construct a Large-Size Simple Integer Cosine Transform for High-Resolution Image and Video Coding. Cybern Syst Anal 52, 956–966 (2016). https://doi.org/10.1007/s10559-016-9898-y

42. Hnativ, L.O. High-Order Integer Cosine Transform: Construction Method and Fast Algorithms for High-Resolution Image and Video Coding. Cybern Syst Anal 53, 615–626 (2017). https://doi.org/10.1007/s10559-017-9964-0

43. Hnativ, L.O. Simple High-Order Integer Cosine Transform for High-Resolution Video Coding. Cybern Syst Anal 54, 490–501 (2018). https://doi.org/10.1007/s10559-018-0050-z

44. Hnativ, L.O. Modified Integer Sine Transforms. Construction Method and Separable Directional Adaptive Transforms for Intra Prediction in Image/Video Coding. Cybern Syst Anal 56, 331–342 (2020). https://doi.org/10.1007/s10559-020-00250-9

45. Hnativ, L.O. Discrete Cosine-Sine Type VII Transform and Fast Integer Transforms for Intra Prediction of Images and Video Coding. Cybern Syst Anal 57, 827–835 (2021). https://doi.org/10.1007/s10559-021-00408-z

46. J. Dong and K.N. Ngan, “A Universal Approach to Developing Fast Algorithm for Simplified Order-16 ICT”, IEEE International Symposium on Circuits and Systems, New Orleans, U.S.A., 27-30 May, 2007, pp. 281-284. doi: 10.1109/ISCAS.2007.378390.

47. Lee B., Kim M., Park C. et al. A 16x16 transform kernel with quantization for (ultra) high definition video coding. ITU-T VCEG, Doc. VCEG-AK13, Yokohoma, Japan, April 2009.
48. C. K. Fong and W. K. Cham, "Simple order-16 Integer Transform for video coding," 2010 IEEE International Conference on Image Processing, Hong Kong, China, 2010, pp. 161-164, doi: 10.1109/ICIP.2010.5651737
49. Gordon S., Marpe D., Wiegand T. “Simplified Use of 8x8 Transforms – Updated Proposal & Results”, Doc JVT-KO28, 11th Meeting: Munich, Germany, 15-19 March, 2004, P.1-22.
50. Ma S., Fan X., Gao W. Low complexity integer transform and High Definition Coding// Proceedings of SPIE 49th Annual Meeting, V. 58. Aug 02-06, 2004, Denver, CO, USA, Bellingham, WA, USA: SPIE Press, 2004. - P. 547-554. https://doi.org/10.1117/12.563808
51. N. Brahimi, T. Bouden, T. Brahimi, and L. Boubchir, "Lossy image compression based on efficient multiplier-less 8-points DCT," Multimedia Systems, vol. 28, no. 1, pp. 171-182, 2022
52. https://www.lcevc.org/
MPEG-5 part 2 LCEVC (Low Complexity Enhancement Video Coding)
Дата звернення 23.10.2023
53. K. Marlapalli, R. S. Bandlamudi, R. Busi, V. Pranav, and B. Madhavrao, "A Review on Image Compression Techniques," in Communication Software and Networks: Springer, 2021, pp. 271-279.
54. A. Moon, J. Kim, J. Zhang, and S. W. Son, "Lossy compression on IoT big data by exploiting spatiotemporal correlation," in 2017 IEEE High Performance Extreme Computing Conference (HPEC), 2017, pp. 1-7.
55. C. Shi, L. Wang, J. Zhang, F. Miao, and P. He, "Remote Sensing Image Compression based on direction lifting-based block transform with content-driven Quadtree coding adaptively," Remote Sensing, vol. 10, no. 7, p. 999, 2018
56. N. Joshi et al., "A review of the application of optical and radar remote sensing data fusion to land use mapping and monitoring," Remote Sensing, vol. 8, no. 1,

p. 70, 2016.

57. I. Blanes, E. Magli, and J. Serra-Sagrista, "A tutorial on image compression for optical space imaging systems," IEEE Geoscience Remote Sensing Magazine, vol. 2, no. 3, pp. 8-26, 2014

58. Y. Ma et al., "Remote sensing big data computing: Challenges and opportunities," Future Generation Computer Systems, vol. 51, pp. 47-60, 2015.

59. K. Kaku, "Satellite remote sensing for disaster management support: A holistic and staged approach based on case studies in Sentinel Asia," International Journal of Disaster Risk Reduction, vol. 33, pp. 417-432, 2019

60. M. Shimoni, R. Haelterman, and C. Perneel, "Hypersectral imaging for military and security applications: Combining myriad processing and sensing techniques," IEEE Geoscience Remote Sensing Magazine, vol. 7, no. 2, pp. 101-117, 2019.

61. Y. Zhang, R. Tian, J. Liu and N. Wang, "Fast rate distortion optimized quantization for HEVC," 2015 Visual Communications and Image Processing (VCIP), Singapore, 2015, pp. 1-4, doi: 10.1109/VCIP.2015.7457854.

62. M. Babic, M. A. Farahani, and T. Wuest, "Image Based Quality Inspection in Smart Manufacturing Systems: A Literature Review," Procedia CIRP, vol. 103, pp. 262- 267, 2021

63. R. Matsumoto, "Introducing the perception-distortion tradeoff into the rate distortion theory of general information sources," IEICE Communications Express, vol. 7, no. 11, pp. 427-431, 2018

64. J. Stankowski, C. Korzeniewski, M. Domański and T. Grajek, "Rate-distortion optimized quantization in HEVC: Performance limitations," 2015 Picture Coding Symposium (PCS), Cairns, QLD, Australia, 2015, pp. 85-89, doi: 10.1109/PCS.2015.7170052.

65. F. Li, S. Krivenko, and V. Lukin, "A Two-step Procedure for Image Lossy Compression by ADCTC With a Desired Quality," in 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies

(DESSERT), 2020, pp. 307-312, doi: 10.1109/DESSERT50317.2020.9125000

66. S. Krivenko, O. Krylova, E. Bataeva, and V. Lukin, "Smart lossy compression of images based on distortion prediction," Telecommunications Radio Engineering, vol. 77, no. 17, 2018.
67. S. Krivenko, F. Li, V. Lukin, B. Vozel, and O. Krylova, "Prediction of visual quality metrics in lossy image compression," in 2020 IEEE 40th International Conference on Electronics and Nanotechnology (ELNANO), 2020, pp. 478-483.
68. F. Li, S. Krivenko, and V. Lukin, "A Two-step Approach to Providing a Desired Visual Quality in Image Lossy Compression," in 2020 IEEE 15th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), 2020, pp. 502-506
69. Ye Y. and Karczewicz M. Improved intra coding. ITU-T SG16Q6. Doc. VCEG-AG11, Shenzhen, China, Oct. 2007.
70. X. Zhao et al., "Transform Coding in the VVC Standard," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 31, no. 10, pp. 3878-3890, Oct. 2021, doi: 10.1109/TCSVT.2021.3087706
71. G. Toderici et al., "Full resolution image compression with recurrent neural networks," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5306-5314
72. L. Zhao, H. Bai, A. Wang, and Y. Zhao, "Multiple description convolutional neural networks for image compression," IEEE Transactions on Circuits Systems for Video Technology, vol. 29, no. 8, pp. 2494-2508, 2018
73. Bouaafia, S., Khemiri, R., Messaoud, S. et al. Deep learning-based video quality enhancement for the new versatile video coding. Neural Comput & Applic 34, 14135–14149 (2022). https://doi.org/10.1007/s00521-021-06491-9
74. Royson Lee, Stylianos I. Venieris, Nicholas D. Lane Deep Neural Network-based Enhancement for Image and Video Streaming Systems: A Survey and Future Directions ACM Computing Surveys Volume 54 Issue 8 pp 1–30 https://doi.org/10.1145/3469094
75. S. Fang, X. Pan, S. Xiang, and C. Pan, "Meta-msnet: Meta-learning based multi-

source data fusion for traffic flow prediction," IEEE Signal Processing Letters, vol. 28, pp. 6-10, 2020.

76. Jain A.K. A fast Karhunen-Loeve transform for a class of stochastic process. IEEE Tran. Commun. 1976. Vol. COM-24. №9. P. 1023-1029
77. Jain A.K. Image coding via nearest neighbors image model IEEE Trans. on Commun.1975. Vol. 23, № 3. P. 318-321.
78. K. McCann, B. Bross, S. Sekiguchi, W.-J. Han, “HM4: High Efficiency Video Coding (HEVC) Test Model 4 Encoder Description”. ITU-T, doc. JCTVC-F802, Torino, IT, July, 2011.
79. Yeo C., Tan Y.H., Li Z. and Rahardja S. Mode-dependent fast separable KLT for block-based intra coding. ITU-T, doc. ICTVC-B024, Geneva, CH, July 2010.
80. Yueh W.C. Eigenvalues of several tridiagonal matrices. Appl. Mathematics E-Notes. 2005. Vol. 5. Apr. P. 66-74.
81. Han J., Saxena A. and Rose K. Towards jointly optimal spatial prediction and adaptive transform in video/image coding. Proc. IEEE Int. Conf. Acoust., Speech, Signal Process (ICASSP), March 2010. P. 726-729.
82. Saxena, A. and Fernandes, F., CE7: Mode-dependent DCT/DST without 4*4 full matrix multiplication for intra prediction. ITU-T, doc. JCTVC- E125, Geneva, CH, Mar. 2011.
83. Chivukula R.K. and Reznik Y.A. Fast computing of discrete cosine and sine transforms of types VI and VII. Proc. SPIE Appl. of Digital Image Process. XXX IV. 2011. Vol. 8135-05. P. 813505-1-9.
84. A. Saxena and F. Fernandes, “Jointly optimal intra prediction and adaptive primary transform,” ITU-T, doc. JCTVC-C108, Guangzhou, China, October 2010.
85. An J., Zhao X., Guo X., Lei S. Non-CE7: Boundary-Dependent Transform for Inter-Predicted Residue/ Document JCTVC- G281, Geneva, CH, Nov. 2011
86. Б.М. Шевчук, В.К. Задірака, Л.О. Гнатів, С.В. Фраєр Технологія багатофункціональної обробки і передачі інформації в моніторингових

мережах. Київ.– Наукова думка. – 2010.–375 С.

87. Гнатів Л.О., Луц В.К. Метод побудови моде-залежного швидкого роздільного цілочисельного ПКЛ для адаптивного кодування зображень і відео. Пр. міжн. наук. конф. “Питання оптимізації обчислень (ПОО-XL), Україна, Крим, Велика Ялта, смт. Кацивелі, вересень 2013. С. 68-69.
88. Saxena A., Fernandes F. C. On fast implementation at 4-point DST type-7 with 5 multiplications. ITU-T, doc. JCTVC – F283, Torino, Italy, July 2011.
89. Saxena A., Fernandes F. C., Alshina E., Chen J. CE7: Mode-dependent DCT/DST for 4x4 chroma blocks. ITU-T, doc. ICTVC-G107, Geneva, CH, Nov. 2011.
90. Fuldseth A., Bjøntegaard G., Sadafale M., at al. CE10: Core transform design for HEVC. ITU-T, doc. JCTVC-G495, Geneva, CH, Nov. 2011.
91. Wang Z., Hunt B.R. The discrete W transform. Appl. Math. and Comput. 1985. Vol. 16, Iss. 1. P. 19-48.
92. Jain A.K. A sinusoidal family of unitary transforms. IEEE Trans. Patt. Anal. and Mach. Intell. 1979. Vol. 1, №4. P. 356-365.
93. Kekre, H.B; Solanki, J.K. Comparative performance of various trigonometric unitary transforms for transform image coding. Int. J. Electronics 1978.Vol. 44, №3, p. 305-315
94. A. Kruger, “Median-cut color quantization,” Dr. Dobb’s Journal, pp. 46-54 and 91-92, Sept.1994.