

Cải tiến thuật toán TZ Search cho tăng tốc mô hình mã hóa H.266/Versatile Video Coding

Bùi Thanh Hương^{1,2}, Nguyễn Quang Sang², Đinh Triều Dương², Chử Đức Trình², Hoàng Văn Xiêm²

¹Khoa Công nghệ Thông tin, trường Đại học Xây dựng

²Khoa Điện tử - Viễn thông, trường Đại học Công nghệ, Đại học Quốc Gia Hà Nội
xiemhoang@vnu.edu.vn

Tóm tắt—Cải tiến các chuẩn mã hóa video đang được quan tâm nhiều trong thời gian gần đây nhằm đáp ứng nhu cầu ngày càng cao của các ứng dụng truyền thông đa phương tiện. Cho đến thời điểm hiện tại, chuẩn mã hóa video mới nhất là chuẩn H.266/VVC (Versatile Video Coding). Với những nỗ lực cải tiến, chuẩn H.266/VVC đạt được lượng bit tiết kiệm lên đến 50% khi so sánh với chuẩn mã hóa video phổ biến H.265/HEVC (High Efficiency Video Coding) trong khi vẫn đảm bảo chất lượng video sau giải mã không đổi. Tuy nhiên, để đạt được hiệu năng mã hóa cao như vậy, chuẩn H.266/VVC yêu cầu thời gian mã hóa gấp 5-30 lần so với chuẩn H.265/HEVC. Nguyên nhân chính đến từ việc phải tìm kiếm khối phù hợp trong một không gian rộng lớn và nhiều trường hợp tìm kiếm hơn. Để giải quyết vấn đề này, bài báo đề xuất một thuật toán cải tiến tìm kiếm nhanh TZ-Search (Test Zone Search) với khả năng tăng tốc độ mã hóa tốt hơn khi dùng trong chuẩn H.266/VVC. Kết quả đánh giá cho thấy, thuật toán TZ-Search cải tiến có thể giúp giảm thời gian mã hóa video H.266/VVC tới 12,6% so với TZ-Search gốc, trong khi vẫn đảm bảo được hiệu năng mã hóa cao.

Từ khóa—Chuẩn H.266/VVC, TZ-Search, ước lượng chuyển động, vùng tìm kiếm

I. GIỚI THIỆU

Trong những năm gần đây, cùng với sự phát triển mạnh mẽ trong nhiều lĩnh vực khoa học công nghệ và sự ra đời của các thiết bị số hiện đại, lượng video cần truyền tải và lưu trữ ngày càng gia tăng. Việc lưu trữ các video thô chưa qua xử lý là bất khả thi bởi chúng yêu cầu một lượng bộ nhớ vô cùng lớn. Các chuẩn mã hóa video mà đi kèm theo là các bộ mã hóa/giải mã hóa video (CODEC: Coder-Decoder) liên tục được ra đời với hiệu năng ngày càng cao nhằm đáp ứng nhu cầu truyền tải, lưu trữ video, góp phần quan trọng vào sự phát triển của các ứng dụng truyền hình thời gian thực (streaming), thực tại ảo (VR: Virtual Reality), thực tại ảo tăng cường (AR: Augmented Reality), 3D-TV,... Trong đó, có thể kể đến các chuẩn mã hóa như H.264/AVC [1], H.265/HEVC [2], AV1 [3] và VP9 [4].

Cuối năm 2020, chuẩn mã hóa video H.266/VVC (Versatile Video Coding) chính thức được thông qua và giới thiệu rộng rãi [5]. Đối tượng chính mà chuẩn mã hóa video này hướng tới là các video với độ phân giải cao, và các nguồn video với định dạng mới như video

360 độ, video chụp từ màn hình (Screen content video). Theo đánh giá hiệu năng công bố ở [6], chuẩn H.266/VVC cho phép người dùng mã hóa video với lượng bit tiết kiệm được khoảng 50% so với chuẩn mã hóa H.265/HEVC trong khi chất lượng video sau giải mã là không đổi.

Mặc dù đạt được hiệu năng mã hóa cao hơn so với chuẩn mã hóa tiền nhiệm của nó, H.266/VVC có độ phức tạp mã hóa tăng từ 5-30 lần so với chuẩn phổ biến H.265/HEVC [7]. Nguyên nhân chính là do VVC cho phép kích thước khối hình dự đoán lớn, số lượng chế độ và hướng dự đoán tăng lên nhằm đáp ứng yêu cầu mã hóa các video có độ phân giải lớn.

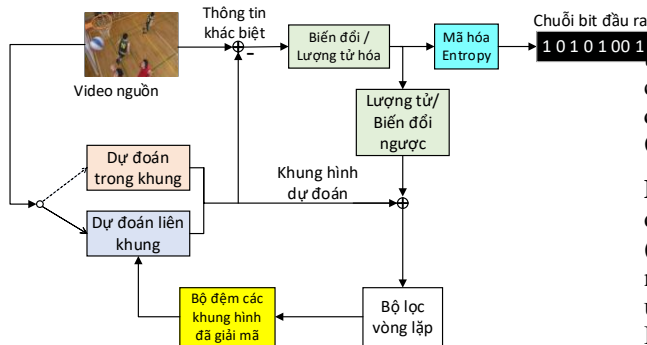
Trong quá trình mã hóa, thời gian dành cho bước ước lượng chuyển động chiếm tới 80% [8]. Do đó, để giảm độ phức tạp mã hóa, thuật toán tìm kiếm nhanh TZ-Search [9] đã được áp dụng để giảm thiểu độ phức tạp của quá trình này. Thuật toán TZ-Search được xây dựng cho phép bộ mã hóa thực hiện dự đoán vector chuyển động bằng cách tìm kiếm điểm dự đoán tối ưu xung quanh điểm dự đoán khởi tạo trên một lưới các điểm được xác định trước thay vì tìm kiếm tất cả các điểm trong vùng tìm kiếm. Việc tối ưu thuật toán TZ-Search luôn là một thách thức cho các nhà nghiên cứu. Với đặc tính đặc thù của video giám sát là thường có tỉ lệ cảnh tĩnh cao và với các đối tượng chuyển động thì thường có vector chuyển động nhỏ, chúng tôi đã tính đến việc thu nhỏ hơn nữa vùng tìm kiếm trong triển khai thuật toán TZ-Search cho loại video giám sát.

Trong phần tiếp theo của bài báo này, chúng tôi trình bày tổng quát về mô hình mã hóa video chuẩn H.266/VVC (phần II) và thuật toán tìm kiếm nhanh TZ-Search. Tiếp đó, phần III và IV lần lượt nêu đề xuất ứng dụng TZ-Search trên các video giám sát với các kết quả thực nghiệm thực tế và các phân tích đánh giá, đây là cơ sở cho phần kết luận chúng tôi nêu trong phần V.

II. TỔNG QUAN VỀ CHUẨN MÃ HÓA VIDEO H.266/VVC

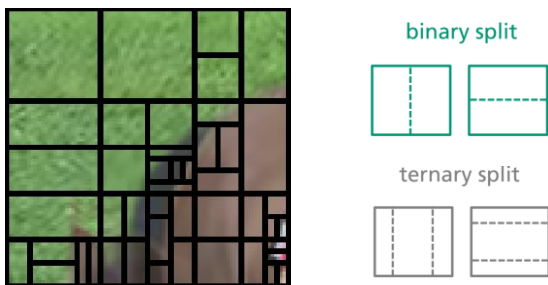
Hình 1 mô tả kiến trúc tổng quan của chuẩn mã hóa video H.266/VVC. Giống như các bộ tiêu chuẩn mã hóa video khác trước đó, VVC là bộ mã hóa theo khối được thiết kế với cấu trúc lai [10] giữa mã hóa dự đoán trong khung và liên khung với mã hóa biến đổi cùng mã hóa entropy.

VVC được thiết kế nhằm hướng tới các đối tượng chính là các video độ phân giải cao (4K, 8K,...), video 360°,... do vậy kích thước khối hình lớn nhất được cho phép trong VVC lên tới 128×128 thay vì 64×64 như trong HEVC [1].



Hình 1: Mô hình mã hóa H.266/VVC

Phân chia khung hình: Sau khi một khung hình video VVC được chia thành các đơn vị mã hóa - CTU (Coding tree unit), các CTU này sau đó tiếp tục được phân chia thành các CU (coding unit), PU (prediction unit) nhỏ hơn, với các kích cỡ khác nhau sao cho phù hợp với nội dung khung hình, chế độ dự đoán. Bên cạnh cấu trúc cây tứ phân (quad tree) đã được sử dụng từ chuẩn mã hóa HEVC, VVC còn cho phép phân chia khối hình bằng cách chia 2 (binary split) hay chia 3 (ternary split) (Hình 2). Cấu trúc phân chia này được gọi là cây tích hợp đa kiểu phân chia đệ quy (nested recursive Multiple-Type Tree (MTT)) với tỉ lệ chiều dài/chiều rộng của khối hình cho phép lên tới 1/16.



Hình 2: Các dạng phân chia đệ quy trong VVC

Dự đoán trong khung: VVC sử dụng kích thước CTU lớn hơn và hình dạng các khối PU sử dụng trong chế độ dự đoán trong khung không còn chỉ là các khối vuông $N \times N$, $2N \times 2N$ và số lượng chế độ dự đoán cũng được tăng lên thành 67 (so với 35 trong H.265/HEVC [2] và 9 trong H.264/AVC [1]). Thay đổi này là để thu được các hướng biên tùy ý của bất kỳ video tự nhiên nào.

Dự đoán liên khung: Đối với dự đoán liên khung, bên cạnh tập các vector dự đoán nâng cao AMVP (advanced Motion Vector prediction) được giới thiệu trong bộ mã hóa HEVC, VVC bổ sung thêm tập vector dự đoán dựa trên lịch sử (History-based Motion Vector Prediction – HMVP) [11].

Biến đổi Cô sin rời rạc (DCT) và lượng tử hóa:

VVC cho phép sử dụng linh hoạt 3 phép biến đổi cô sin rời rạc bao gồm DCT-II, DCT-VIII, DST-VII [10]. Việc này giúp giảm thiểu chi phí biến dạng. Hơn nữa, đối với video có độ phân giải cao hơn (1080p và 4K), việc cho phép chuyển đổi kích thước khối lớn (lên đến 28×28) rất hữu ích trong việc loại bỏ hệ số biến đổi tần số cao. Hệ số lượng tử (QP) cũng được mở rộng cho phép lên tới 63 và một ánh xạ QP linh hoạt hơn được áp dụng để chuyển đổi QP độ xám (luma) sang QP màu sắc (chroma).

Bộ lọc nhiễu khối và bù mẫu thích ứng: VVC sử dụng ba bộ lọc riêng biệt, trong đó có hai bộ lọc lặp (In-Loop) đầu tiên đã triển khai trong HEVC là Bộ lọc nhiễu khối (DBF: Deblocking Filter) và Bù mẫu thích ứng (SAO: Sample Adaptive Offset). Bộ lọc thứ ba là Bộ lọc vòng lặp thích ứng (ALF: Adaptive Loop Filter) đã được cập nhật trong chuẩn VVC. Hơn nữa, trong các bộ lọc In-Loop của VVC, các bộ lọc hình dạng kim cương 7×7 và 5×5 được áp dụng cho các thành phần độ xám và các màu sắc theo thứ tự tương ứng.

Mã hóa nhị phân thích ứng: Thông tin dư thừa hay thông tin khác biệt (là thông tin sai khác giữa khung hình hiện tại và khung hình dự đoán tương ứng) được mã hóa bằng công cụ mã hóa số học, nhị phân tương thích ngữ cảnh (Context Adaptive Binary Arithmetic Coding – CABAC) đã được chứng minh mang lại hiệu năng cao ở chuẩn mã hóa HEVC.

III. ĐỀ XUẤT THUẬT TOÁN TZ-SEARCH CẢI TIẾN

Để tìm kiếm vị trí khối phù hợp cho khối ảnh hiện tại trong thực hiện phép dự đoán khung hình tiếp theo, thuật toán tìm kiếm nhanh TZ-Search đã được thông qua ở chuẩn H.266/VVC. Với TZ-Search, thay vì thực hiện tìm kiếm trên toàn bộ các điểm ảnh trong vùng tìm kiếm, quá trình ước lượng chuyển động của bộ mã hóa chỉ được thực hiện dựa trên một số điểm ảnh có khả năng phù hợp nhất. Hình 3 mô tả thuật toán TZ-Search, trong đó có nhấn mạnh nội dung cải tiến (khối màu vàng).

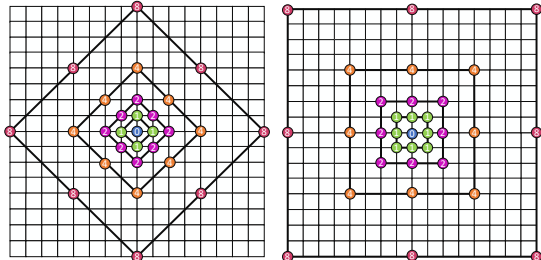
Thuật toán TZ-Search có thể được thực hiện thông qua các bước sau:

- **Bước 1: Khởi tạo điểm tìm kiếm**

Tại bước này, tập các vector chuyển động được thiết lập bao gồm vector từ các điểm bên trái, phía trên, phía trên bên phải của khối ảnh tương ứng với khung tham chiếu, từ đó thiết lập vector trung vị của 3 vector này và vector 0. Nhìn chung, điểm được lựa chọn để bắt đầu quá trình thực hiện tìm kiếm dựa trên vector chuyển động của điểm đó với sự sai khác thấp nhất (SAD: Sum of Absolute Difference là nhỏ nhất) và các điểm này còn được gọi là điểm tối ưu, trong đó vector trung vị giúp xác định điểm tìm kiếm khởi tạo đầu tiên.

- **Bước 2: Tìm kiếm theo mô hình kim cương hoặc hình vuông**

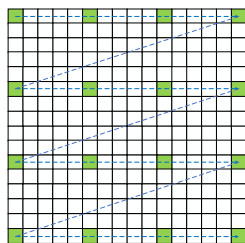
Ở bước này, trước hết vùng tìm kiếm (Search Range) được xác định và mô hình tìm kiếm kim cương hoặc hình vuông được lựa chọn (Hình 4). Các điểm tìm kiếm là các điểm tại đỉnh và trung điểm của các cạnh hình vuông hoặc hình thoi với khoảng cách thay đổi từ 1 đến giá trị vùng tìm kiếm và ở dạng hàm mũ của cơ số 2. Sau khi tìm được điểm tối ưu, bộ mã hóa tiếp tục ước lượng chuyển động tại 2 điểm liền kề của điểm đó.



Hình 4: Mô hình kim cương và hình vuông

• **Bước 3: Tìm kiếm Raster**

Xét khoảng cách từ điểm khởi tạo đến điểm tối ưu hay chính là khoảng cách tối ưu (BestDistance) với các trường hợp cụ thể sau: Nếu khoảng cách tối ưu tại bước 2 bằng 0, quá trình tìm kiếm kết thúc. Nếu khoảng cách tối ưu nằm trong khoảng $(1, iRaster)$ thì bỏ qua bước 3 này, chuyển sang thực hiện bước 4, tức là thực hiện tinh chỉnh tìm kiếm Raster/Star. Còn nếu khoảng cách tối ưu lớn hơn hoặc bằng khoảng cách $iRaster$, bộ mã hóa tiến hành tìm kiếm Raster với bước nhảy được đặt bằng thông số $iRaster$ (Hình 5). Cuối cùng, nếu khoảng cách này bằng 1 thì gán lại khoảng cách này bằng 0 và chuyển sang thực hiện kiểm tra hai điểm lân cận.



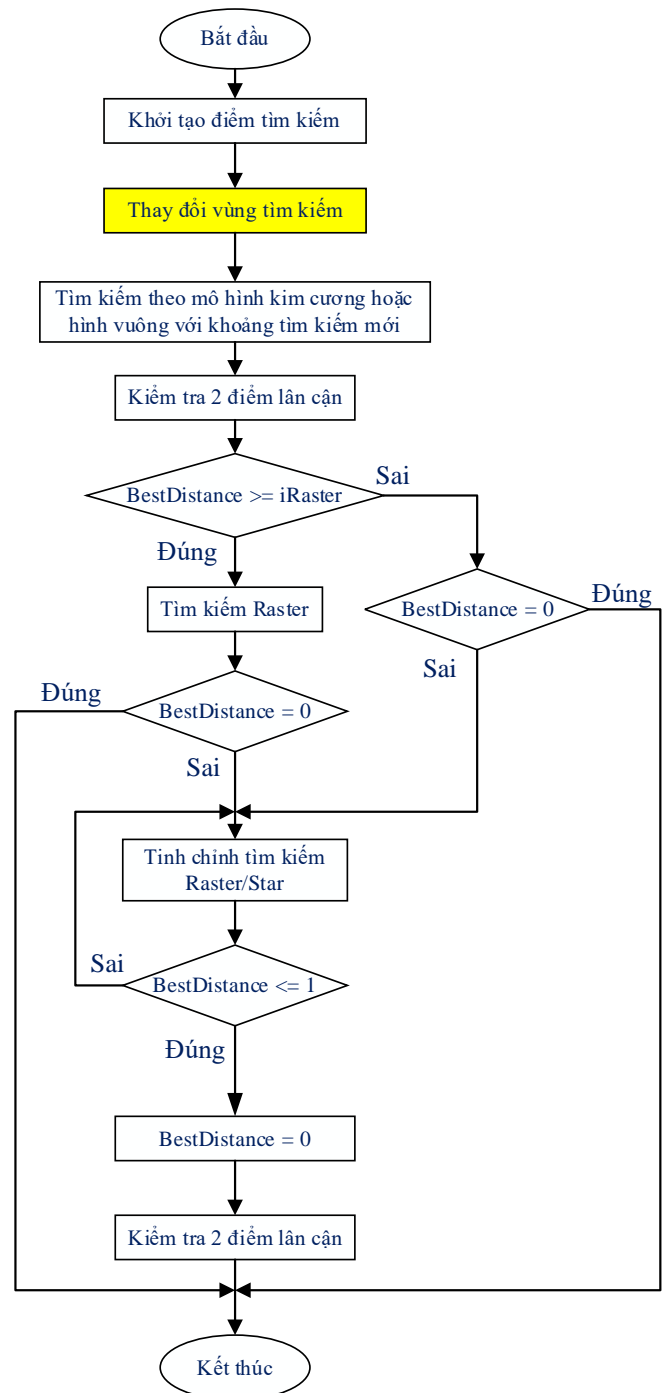
Hình 5: Mô hình tìm kiếm Raster

• **Bước 4: Tinh chỉnh tìm kiếm Raster/Star**

Điểm tối ưu tìm được tại bước 3 được đặt thành điểm khởi tạo tìm kiếm mới cho quá trình tìm kiếm theo mô hình kim cương hoặc hình vuông. Các điểm tối ưu được cập nhật liên tục sau mỗi vòng tìm kiếm, nếu khoảng cách tối ưu bằng 1 thì đặt lại khoảng cách này bằng 0 và sau đó kiểm tra 2 điểm lân cận. Quá trình này kết thúc khi tìm được điểm có khoảng cách tối ưu = 0.

Trong dự đoán liên khung, quá trình ước lượng chuyển động đã thực hiện việc khai thác đặc tính chung của các loại video là giữa các khung liền kề có nhiều thông tin có tính tương quan cao về chuyển động. Cần nhắc thêm đặc tính riêng biệt của video giám sát là các

vùng chuyển động của đối tượng trong video giám sát thường nhỏ, do đó vùng tìm kiếm triển khai trong thuật toán TZ-Search có thể thu nhỏ hơn khi thực hiện mã hóa video giám sát. Hình 6 sau đây mô tả đề xuất của chúng tôi về việc giảm kích thước vùng tìm kiếm trong thuật toán TZ-Search và các kết quả thực nghiệm thực tế cùng các đánh giá, phân tích cụ thể.



Hình 3: Thuật toán TZ-Search đề xuất

Đề xuất thu nhỏ vùng tìm kiếm thích ứng trong TZ-Search

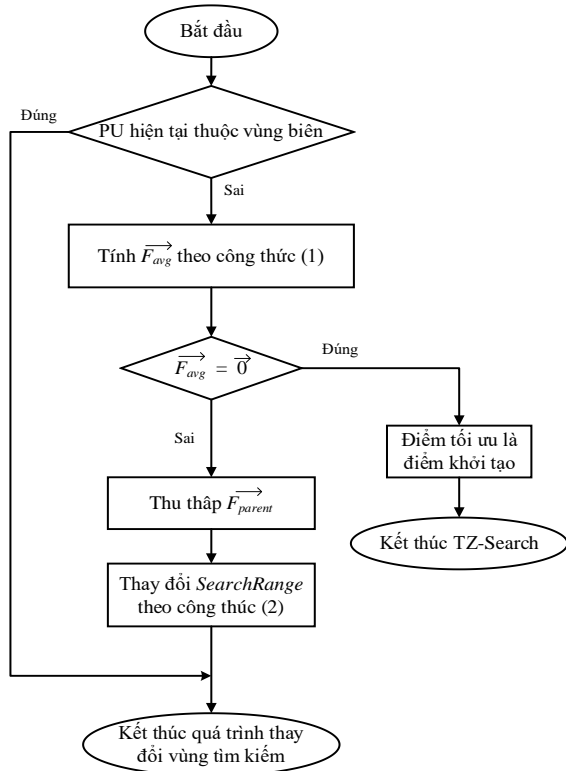
Kích thước vùng tìm kiếm có ảnh hưởng trực tiếp tới số lượng điểm tìm kiếm trong mã hóa liên khung. Do vậy, để tăng tốc độ mã hóa, chúng tôi đề xuất thay đổi vùng tìm kiếm dựa theo các đặc tính chuyển động của khối ảnh. Cụ thể, dựa vào vector chuyển động của các khối ảnh bên trái, phía trên và góc trên bên trái của khối ảnh hiện tại, thực hiện tính vector trung bình theo công thức như sau:

$$\vec{F}_{avg} = \frac{\vec{F}_{Top} + \vec{F}_{Left} + \vec{F}_{TopLeft}}{3} \quad (1)$$

Nếu $\vec{F}_{avg} = \vec{0}$ thì điểm khởi tạo (tiếp theo) chính là điểm tối ưu, vậy thực hiện dừng quá trình tìm kiếm theo thuật toán TZ-Search. Nếu $\vec{F}_{avg} \neq \vec{0}$, lúc này, thực hiện thu nhỏ vùng tìm kiếm dựa vào thông tin của vector chuyển động của block cha \vec{F}_{parent} theo công thức sau:

$$SearchRange = \left\lfloor \frac{\vec{F}_{avg} + \vec{F}_{parent}}{2} \right\rfloor \quad (2)$$

Toàn bộ quá trình thu nhỏ vùng tìm kiếm thích ứng, đề xuất sử dụng trong TZ-Search có thể được mô tả như hình 6.



Hình 6: Thuật toán thay đổi vùng tìm kiếm trong TZ-Search

IV. MÔ PHỎNG VÀ ĐÁNH GIÁ KẾT QUẢ

A. Điều kiện mô phỏng, đánh giá

Để đánh giá hiệu quả của phương pháp đề xuất, chúng tôi thực hiện việc mô phỏng và kiểm thử trên 08 chuỗi video giám sát có độ phân giải 720×576 và 1600×1200 [12, 13] với chế độ mã hóa Random Access. Các chuỗi video và thông số cụ thể được thể hiện trong Bảng 1 và Bảng 2.

BẢNG 1: CÁC ĐIỀU KIỆN CHẠY KIỂM THỬ

Chuỗi video có độ phân giải 720×576	Bank
	Campus
	Classover
	Crossroad
	Office
Chuỗi video có độ phân giải 1600×1200	Overbridge
	Intersection
Tốc độ khung hình và số khung hình mã hóa	Mainroad
	30Hz, 128 khung hình
Kích thước nhóm ảnh GOP (group of picture) mã hóa	16
Hệ số lượng tử (QP: quantization parameters)	QP = {22, 27, 32, 37}
Kích thước vùng tìm kiếm mặc định	384
Chu kỳ lặp lại khung I (Intra period)	32

BẢNG 2: CẤU HÌNH PHẦN CỨNG

Bộ vi xử lý	Intel® Core™ i7-4800MQ @2.7 GHz
RAM	8.00 GB
System	Win 10, 64-bit
Môi trường kiểm thử	Microsoft Visual Studio 2017 Community

Thời gian mã hóa video được đo cho bộ mã hóa H.266/VVC sử dụng thuật toán TZ-Search gốc ($T_{TZ-Search_VVC}$) và thuật toán TZ-Search cải tiến ($T_{TZ-Search_dexuat}$) cho các giá trị lượng tử khác nhau. Mức độ giảm thiểu về mặt thời gian được xác định theo công thức sau:

$$\Delta T = \frac{T_{TZ-Search_dexuat} - T_{TZ-Search_VVC}}{T_{TZ-Search_VVC}} \times 100 \quad (3)$$

Hiệu năng mã hóa với chuẩn H.266/VVC thường được xác định thông qua hai hình thức: sử dụng biểu đồ độ biến dạng – tốc độ bit (RD) hoặc giá trị Bjontegaard Rate [14]. Trong bài báo này, chúng tôi trình bày cả hai hình thức đánh giá trên.

B. Đánh giá thời gian tăng tốc độ mã hóa

Bảng 3 tổng hợp kết quả so sánh độ phức tạp nén video trong trường hợp dùng TZ-Search đề xuất với TZ-Search trong VVC và kết quả giảm hiệu năng mã hóa BD-Rate.

Với mục tiêu giảm độ phức tạp của thuật toán tìm kiếm nhanh TZ-Search, đề xuất thay đổi kích thước

vùng tìm kiếm trong mã hóa video giám sát đã giúp tiết kiệm được thời gian mã hóa trên tất cả các mức lượng tử và độ phân giải, kết quả thực nghiệm cho thấy thời gian mã hóa giảm từ 7,25% đến 12,62% (trung bình đạt khoảng 9,3%) trong khi số lượng bit cần dùng gần như không tăng. Thêm nữa, trong số các chuỗi video được thử nghiệm, chuỗi video Office tiết kiệm được thời gian mã hóa cao nhất, đạt trung bình 10,5%.

BẢNG 3: KẾT QUẢ GIẢM THỜI GIAN NÉN VIDEO ($\Delta T\%$) VÀ CHI PHÍ HIỆU NĂNG MÃ HÓA (BD-RATE) SO VỚI TZ SEARCH GỐC

Tên video	QP				Trung Bình	BD-Rate
	22	27	32	37		
Bank	-10.90	-9.29	-9.22	-9.96	-9.84	0,50
Campus	-9.15	-8.91	-8.91	-8.42	-8.85	0,61
Classover	-7.72	-7.83	-8.46	-7.25	-7.82	0,12
Crossroad	-7.84	-8.70	-7.18	-7.98	-7.93	0,88
Office	-9.68	-10.41	-10.58	-11.32	-10.50	0,11
Overbridge	-10.66	-9.83	-9.80	-9.00	-9.82	0,83
Intersection	-10.51	-8.36	-9.01	-12.62	-10.13	0,59
Mainroad	-11.52	-9.83	-9.15	-7.63	-9.53	-0,02
Trung Bình	-9.75	-9.15	-9.04	-9.27	-9.30	0,45

V. KẾT LUẬN

Trong bài báo này, chúng tôi đã nghiên cứu tổng quan về các cải tiến trong chuẩn mã hóa video mới nhất, vừa được chính thức thông qua trong tháng 7/2020, qua đó, chúng tôi tập trung khai thác thuật toán tìm kiếm nhanh TZ-Search với cải tiến giảm độ phức tạp của thuật toán bằng cách xác định giảm vùng tìm kiếm. Qua thực nghiệm, các kết quả khẳng định tính đúng đắn của đề xuất: đã giảm được khoảng 7-12% thời gian mã hóa video giám sát mà lượng bitrate gần như không bị ảnh hưởng (tăng trung bình 0,45%). Hướng nghiên cứu tiếp theo sẽ tập trung vào đánh giá mức độ ảnh hưởng của từng khối trong TZ-Search để cải tiến cả độ chính xác tìm kiếm lẫn khả năng tăng tốc độ tìm kiếm.

VI. LỜI CẢM ƠN

Nghiên cứu này được tài trợ bởi Quỹ Phát triển khoa học và công nghệ Quốc gia (NAFOSTED) trong đề tài mã số 102.01-2020.15

TÀI LIỆU THAM KHẢO

- [1] T. Wiegand, G.J. Sullivan, G. Bjontegaard, A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 7, pp. 560–576, 2003.
- [2] G.J. Sullivan, et al., "Overview of the High Efficiency Video Coding (HEVC) Standard", IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 12, pp. 1649-1668, 2012.
- [3] Y. Chen, et al., "An Overview of Core Coding Tools in the AV1 Video Codec", Picture Coding Symposium (PCS), 2018.
- [4] D. Mukherjee, J. Bankoski, A. Grange, J. Han, J. Koleszar, P. Wilkins, Y. Xu, R. Bultje, "The latest open-source video codec VP9 – An overview and preliminary results", Picture Coding Symposium (PCS), 2013.
- [5] Fraunhofer HHI is proud to present the new state-of-the-art in global video coding: H.266/VVC brings video transmission to new speed. [Online]. Available: <https://newsletter.fraunhofer.de/viewonline2/17386/465/11/14SHcBTv/V44RELLZBp/1>
- [6] N. Sidaty, W. Hamidouche, O. Déforges, P. Philippe, J. Fournier, "Compression Performance of the Versatile Video Coding: HD and UHD Visual Quality Monitoring", Picture Coding Symposium (PCS), Ningbo, China, 2019.
- [7] F. Bossen, X. Li and K. Suehring, "AHG report: Test model software development (AHG3)", document JVET-R0003, ITU-T/ISO/IEC Joint Video Experts Team (JVET), 2020.
- [8] N. C. Vayalil, M. Paul, Y. Kong, "A novel angle-restricted Test Zone Search algorithm for performance improvement of HEVC", ICIP, 2017.
- [9] X. Li, R. Wang, W. Wang, Z. Wang, S. Dong, "Fast Motion Estimation Methods for HEVC", IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, 2014.
- [10] J. Chen, Y. Ye, S. H. Kim, "Algorithm description for Versatile Video Coding and Test Model 8 (VTM 8)", JVET-Q2002-v3, 2020.
- [11] L. Zhang, K. Zhang, H. Liu, H. C. Chuang, Y. Wang, J. Xu, P. Zhao, D. Hong, "History-based Motion Vector Prediction in Versatile Video Coding", Data Compression Conference (DCC), 2019.
- [12] W. Gao, Y. Tian, T. Huang, S. Ma, and X. Zhang, "IEEE 1857 standard empowering smartvideo surveillance systems", IEEE Intelligent Systems, 2013.
- [13] PKU-SVD-A. [Online]. Available: <http://mlg.idm.pku.edu.cn/resources/pku-svd-a.html>
- [14] G. Bjontegaard, "Calculation of Average PSNR Differences between RD Curves", Document VCEG-M33, 13th ITU-T VCEG Meeting, Austin, TX, USA, 2001.