ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

VŨ DUY KHƯƠNG

PHÁT TRIỂN THUẬT TOÁN NỘI SUY NHẰM TĂNG CƯỜNG CHẤT LƯỢNG VIDEO TRONG 3D-HEVC

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

HÀ NỘI - 2016

ĐẠI HỌC QUỐC GIA HÀ NỘI

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

VŨ DUY KHƯƠNG

PHÁT TRIỂN THUẬT TOÁN NỘI SUY NHẰM TĂNG CƯỜNG CHẤT LƯỢNG VIDEO TRONG 3D-HEVC

Ngành : Công Nghệ Thông Tin

Chuyên ngành : Kỹ Thuật Phần Mềm - 60.48.01.03

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS.TS. Lê Thanh Hà

TS. Đinh Triều Dương

HÀ NỘI - 2016

LỜI CAM ĐOAN

Tôi xin cam đoan : Luận văn "**Phát triển thuật toán nội suy nhằm tăng cường chất lượng video trong 3D-HEVC**" là công trình nghiên cứu riêng của tôi, không sao chép của ai. Các số liệu trong luận văn được sử dụng trung thực. Kết quả nghiên cứu được trình bày trong luận văn này chưa từng được công bố tại bất kỳ công trình nào khác

Hà Nội, Ngày…..tháng….năm 2016

Tác giả

Vũ Duy Khương

LỜI CÁM ƠN

Luận văn của tôi không thể được hoàn thành nếu không được sự giúp đỡ, hỗ trợ và khuyến khích của nhiều người, đặc biệt tôi thực sự biết ơn đến các thầy hướng dẫn tôi: PGS.TS. Lê Thanh Hà, TS. Đinh Triều Dương. Các thầy đã cho tôi rất nhiều lời khuyên có giá trị trong phương pháp nghiên cứu, văn phong viết, kỹ năng trình bày...Tôi thực sự cảm thấy rất may mắn là một trong những học sinh của thầy.

Tôi muốn cảm ơn tất cả bạn bè của tôi, bạn bè trong phòng thí nghiệm tương tác người máy HMI về các cuộc thảo luận hữu ích cuả họ về chủ đề nghiên cứu của tôi

Tôi xin được gửi lời cảm ơn đến tất cả quý thầy cô đã giảng dạy trong chương trình Cao học Công nghệ thông tin - Trường Đại học công nghệ, những người đã truyền đạt cho tôi những kiến thức hữu ích về Công nghệ làm cơ sở cho tôi thực hiện tốt luận văn này.

Hà Nội, Ngày….tháng….năm 2016

Học viên

Vũ Duy Khương

MỤC LỤC

DANH MỤC KÝ HIỆU, TỪ VIẾT TẮT

Số	Thuật Ngữ	Giải Thích
1	TV	Television
2	3D	Three Dimension
3	MVD	Multiview Video plus Depth
4	MVV	Multi Vew Video
5	DIBR	Depth Image Based Rendering
6	MPEG	Moving Pictures Experts Group
7	VSRS	View Synthesis Reference Software
8	HEVC	High Efficiency Video Coding
9	MVF	Motion View Field
10	DIBR	Depth Image Based Rendering
11	PSNR	Peak Signal to Noise Ratio
12	SWA	Spiral weighted average algorithm

DANH MỤC HÌNH VẼ

DANH MỤC BẢNG BIỂU

MỞ ĐẦU

Các kỹ thuật 3D video đang ngày càng mang lại những trải nghiệm thực tế đối với người sử dụng. Vì vậy hầu hết các bộ phim 3DTV [1] và 3D hiện nay là các hiển thị thực thể 3D, các nội dung 3D sẵn có đều ở định dạng thực thể 3D. Trong trường hợp này, các vấn đề này phát sinh là do góc nhìn hẹp và yêu cầu người xem phải đeo kính để xem các nội dung 3D. Để giải quyết vấn đề này, việc nghiên cứu hiển thị thực thể tự động và FTV [2] được đặt ra. Hiển thị thực thể tự động cung cấp nhận thức chiều sâu 3D mà không cần phải đeo kính bằng cách cung cấp đồng thời 1 số lượng hình ảnh khác nhau. FTV cho phép người xem có thể xem ở bất cứ điều kiện xem nào. Tuy nhiên, trong các trường hợp đó, chúng ta cần nhiều băng thông hơn để truyền tải và cần lưu trữ dữ liệu lớn cũng như là các chi phí đáng kể cho việc thiết đặt nhiều camera

Nhìn chung, hệ thống hiển thị tự động thực thể 3D cần nhiều hình ảnh đầu vào. Có 3 phương pháp thu thập hình ảnh đa điểm. Đầu tiên, chúng ta có thể có hình ảnh đa điểm bằng cách sử dụng nhiều camera như số quan sát được yêu cầu. Tuy nhiên, trong trường hợp này, việc đồng bộ hóa và tính toán các camera này là rất khó khăn. Lựa chọn tiếp theo là sử dụng 1 hệ thống camera có thể có được một hình ảnh màu với bản đồ độ sâu tương ứng với ảnh màu đó và tổng hợp lên hình ảnh trung gian ảo từ dữ liệu thu được. Lựa chọn cuối cùng là ước lượng được độ chênh lệch từ những hình ảnh thu được từ 2 camera màu tổng hợp lên hình ảnh. MPEG coi TV như là dịch vụ phương tiện truyền thông 3D hứa hẹn nhất và đã bắt đầu chuẩn hóa theo tiêu chuẩn quốc tế từ năm 2002. Nhóm 3DV [3] trong MPEG đang làm việc theo 1 tiêu chuẩn có thể được sử dụng để sử dụng cho 1 loạt các định dạng hiển thị 3D. 3DV là 1 framework mới bao gồm hiển thị thông tin đa điểm video và thông tin độ sâu để hỗ trợ thế hệ tiếp theo. Do đó, việc ước lượng chiều sâu và quá trình tổng hợp là 2 quá trình quan trọng trong 3DV vì vậy chúng ta cần 1 thuật toán chất lượng cao. Chúng ta có thể sử dụng giới hạn số lượng hình ảnh camera để sinh ra nhiều hình ảnh bằng cách sử dụng thuật toán DIBR [4] (depth image based rendering).

DIBR là 1 trong những kỹ thuật phổ biến được sử dụng để biểu diễn các khung hình ảo. Một hình ảnh màu và bản đồ độ sâu cho mỗi điểm ảnh tương ứng của nó được

sử dụng cho tổng hợp 3D dựa trên nguyên tắc hình học. Tuy nhiên, việc trích xuất chính xác độ lệch hay bản đồ độ sâu tiêu tốn nhiều thời gian và rất khó khăn. Hơn nữa, sẽ tồn tại các hố và nhiễu biên (boundary noise) [5] trong hình ảnh tổng hợp do các occlusion và sai số độ lệch. Các nhiễu biên xảy ra do không chính xác biên giữa độ sâu và vân ảnh trong suốt quá trình tổng hợp 3D và điều này đã gây ra những điểm bất thường trong khung hình ảo được sinh ra. Ngoài ra, các hố thông thường (common-holes) [6] cũng được tạo ra trong khi tổng hợp lên khung hình ảo. Các hố thông thường này được khắc phục dựa trên thông tin các vùng xung quanh hố. Tuy nhiên, việc khắc phục các hố thông thường là khó khăn về quá trình thực hiện và về mặt thị giác. Do đó chúng ta cần cách mới để thực hiện lấp đầy các hố này với hiệu suất cao nhất. Để lấp đầy các hố thông thường, phương pháp nội suy tuyến tính và phương pháp inpainting được đề xuất. Phương pháp inpainting [7] ban đầu được sử dụng để khôi phục các vùng hư hại của ảnh bằng cách ước lượng giá trị từ thông tin màu sắc được cung cấp. Phương pháp này thường được dùng để khắc phục các vùng hư hại của ảnh. Phương pháp nội suy tuyến tính là việc thêm hoặc trừ đi các giá trị điểm ảnh ở vị trí đối diện xung quanh vùng các hố. Tiến trình này yêu cầu ít thời gian nhưng chất lượng hiện tại của các hố là không hiệu quả. Chính vì vậy, việc nghiên cứu một phương pháp nội suy mới nhằm nâng cao chất lượng video là điều cần thiết. Thuật toán Hole filling SWA là thuật toán dựa trên trọng số trung bình về độ sâu và sử dụng các thông tin về gradient để lấp đầy các hố

trong video. Thuật toán này đã đáp ứng yêu cầu cấp thiết, nhằm nâng cao chất lượng video thực tế.

Trong luận văn này, luận văn sẽ nghiên cứu các vấn đề về 3DTV, TV, các phần mềm tham chiếu, cài đặt thuật toán Hole filling SWA (Spiral weighted average algorithm) [6] và cuối cùng so sánh hiệu suất so với các thuật toán Hole filling khác.

CHƯƠNG 1: ĐẶT VẤN ĐỀ

1.1. LÝ DO CHỌN ĐỀ TÀI

Để cung cấp những trải nghiệm 3D thực, chúng ta cần nhiều video được chụp từ các điểm quan sát khác nhau. Nhưng thực tế cho thấy, gần như là không thể để chụp và chuyển một lượng lớn các khung hình được yêu cầu. Kết quả là chúng ta cần một kỹ thuật biểu diễn để tạo ra một nội dung thích hợp cho các ứng dụng này. Thiết bị đóng vai trò quan trọng nhất là FTV [2]. Thực tế cho thấy hình ảnh 3D được tổng hợp lên từ các camera cho kết quả không được cao như mong đợi. Tồn tại các hố và nhiễu biên (boundary noise) trong hình ảnh tổng hợp do các occlusion và sai số độ lệch. Các nhiễu biên xảy ra do không chính xác biên giữa độ sâu và vân ảnh trong suốt quá trình tổng hợp 3D và điều này đã gây ra những điểm bất thường trong khung hình ảo được sinh ra. Tuy nhiên, việc khắc phục các hố thông thường là khó khăn về quá trình thực hiện và về mặt thị giác. Do đó chúng ta cần cách mới để thực hiện lấp đầy các hố này với hiệu suất cao nhất. Đã có rất nhiều thuật toán, ứng dụng được đề xuất. Tuy nhiên, mỗi thuật toán, ứng dụng lại có ưu nhược điểm hạn chế riêng. Chính vì vậy, nhằm nâng cao chất lượng đầu ra cho chất lượng khung hình 3D tổng hợp lên. Việc tìm ra thuật toán tối ưu là cấp bách. Trên cơ sở thực tiễn này. Luận văn trình bày một thuật toán nội suy mới tối ưu nhằm nâng cao chất lượng hình ảnh 3D. Thuật toán nội suy mà luận văn đề cập ở đây là thuật toán Hole filling SWA [6] sẽ được trình bày chi tiết ở Chương 3.

1.2. MỤC TIÊU CỦA LUẬN VĂN

Mục tiêu của luận văn là nghiên cứu kỹ thuật DIBR dùng trong 3DTV và tập trung phân tích tìm hiểu thuật toán Hole filling SWA. Nghiên cứu, so sánh các thuật toán Hole filling. Cài đặt và thử nghiệm thuật toán nhằm đánh giá khả năng loại bỏ các nhiễu biên, tính hiệu quả của thuật toán trong việc nội suy nhằm loại bỏ các hố trong khung hình ảo dựa trên thuật toán trọng số trung bình đường xoắn ốc và thuật toán gradient để nhằm tăng cường chất lượng khung hình tổng hợp.

1.3. CẤU TRÚC LUẬN VĂN

Luận văn được tổ chức như sau:

Chương 1: Đặt vấn đề, đề xuất, trình bày luận văn, các vấn đề liên quan, mục tiêu nghiên cứu, các đóng góp của luận văn

Chương 2: Trình bày các khái niệm cơ bản liên quan đến vấn đề nghiên cứu như **FTV**, **3DTV, VSRS, HEVC**,…

Chương 3: Trình bày thuật toán Hole filling SWA

Chương 4: Trình bày kết quả thí nghiệm, đề xuất, chỉ ra hướng nghiên cứu

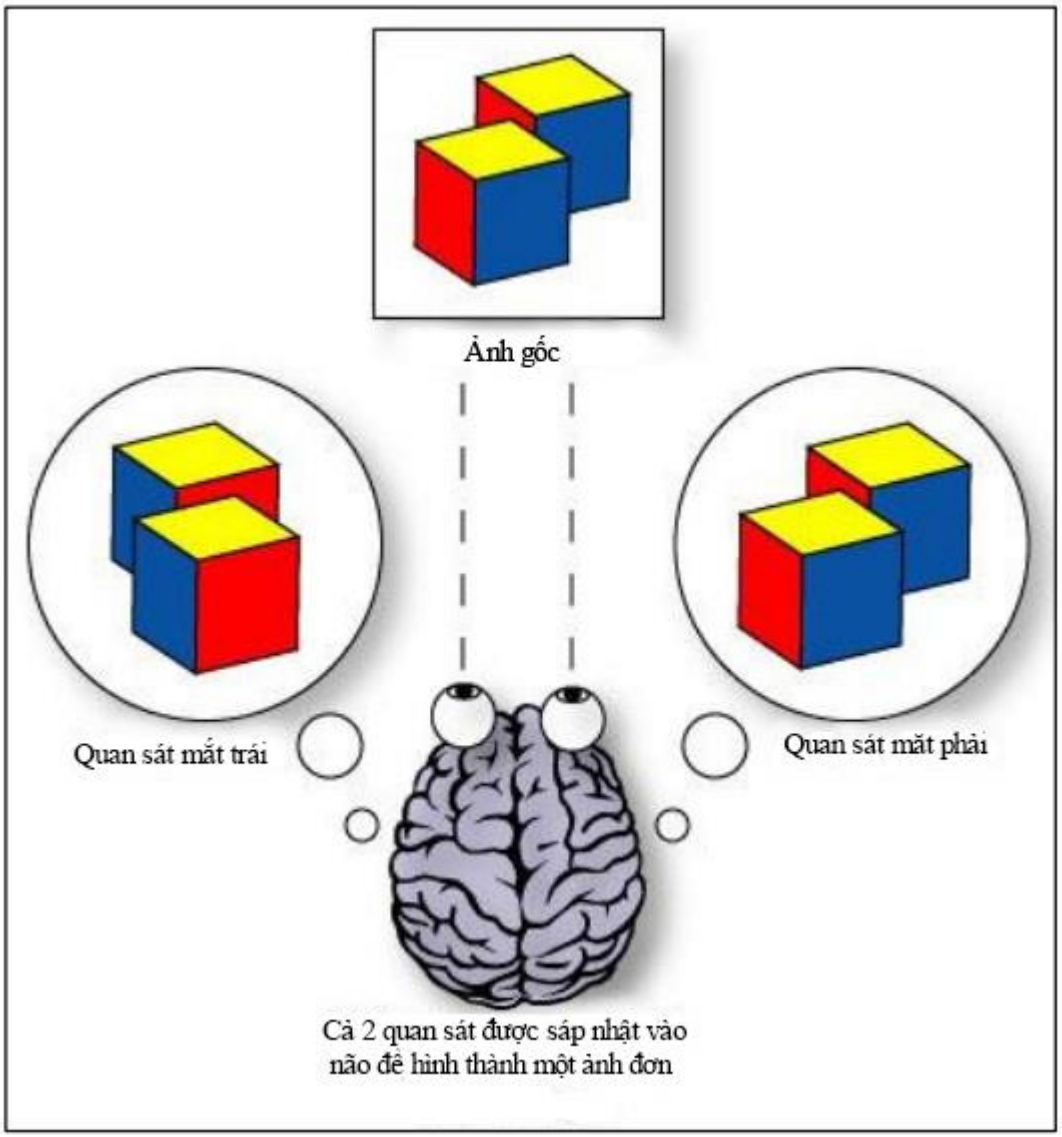
CHƯƠNG 2: CÁC KHÁI NIỆM CƠ BẢN

Chương này giới thiệu các khái niệm cơ bản trong lĩnh vực video coding đặc biệt có sự tham khảo chuẩn HEVC, mở rộng mã hóa Multi-view video và tổng hợp quan sát dựa trên chiều sâu. Chương này bắt đầu với cái nhìn tổng quan về video. Bao gồm 2.1 giới thiệu về các ứng dụng video giả lập 3D. Mục **2.1.1** giới thiệu về **Tivi 3D**. **Tivi Free VewPoint** được giới thiệu trong Mục **2.1.2**. Các định dạng biểu diễn video 3D được giới thiệu trong Mục **2.2**. Mục **2.2.1** Giới thiệu về **MVV** và **MVVD**, **2.2.2** nói về bản đồ độ sâu. Cuối cùng, biểu diễn dựa trên ảnh độ sâu được giới thiệu trong mục **2.3**, có 3 bước: Tổng hợp 3D, sáp nhập khung hình và hole filling các vùng disocclusion

2.1. CÁC ỨNG DỤNG VIDEO GIẢ LẬP 3D

2.1.1. TIVI 3D (3DTV)

Con người chúng ta có hai mắt, nằm gần nhau và bên cạnh nhau. Mỗi mắt có một quan sát khu vực nhìn từ một góc khác nhau. Não chúng ta nhận các hình ảnh từ hai mắt và kết hợp chúng bằng những điểm tương đồng. Bên cạnh đó, sự khác biết nhỏ nhất giữa hai hình ảnh được giải thích bằng thông tin về độ sâu. Quá trình này tạo ra một khung hình 3D: một với chiều cao, một với chiều rộng và với chiều sâu. Thị giác của con người được goi là thị giác lập thể. Nguyên tắc thị giác của người được minh họa trong Hình 2.1. Nguyên tắc này có thể được áp dụng đối với công nghệ hiển thị video. Nếu màn hình cung cấp những cái nhìn đúng đắn để mắt tương thích, nó có thể bắt chước điều kiện thị giác con người một cách tự nhiên và sự khác biệt trong hình ảnh lập thể có thể được chuyển đổi thành chiều sâu. Những hình ảnh lập thể tương ứng với mắt có thể đạt được theo nhiều cách khác nhau chẳng hạn như đeo kính đặc biệt có thể lọc được những hình ảnh chính xác cho mắt nhìn chính xác như trong hiển thị lập thể. Các kỹ thuật khác sử dụng các thành phần quang học được tích hợp trong màn hình khác.

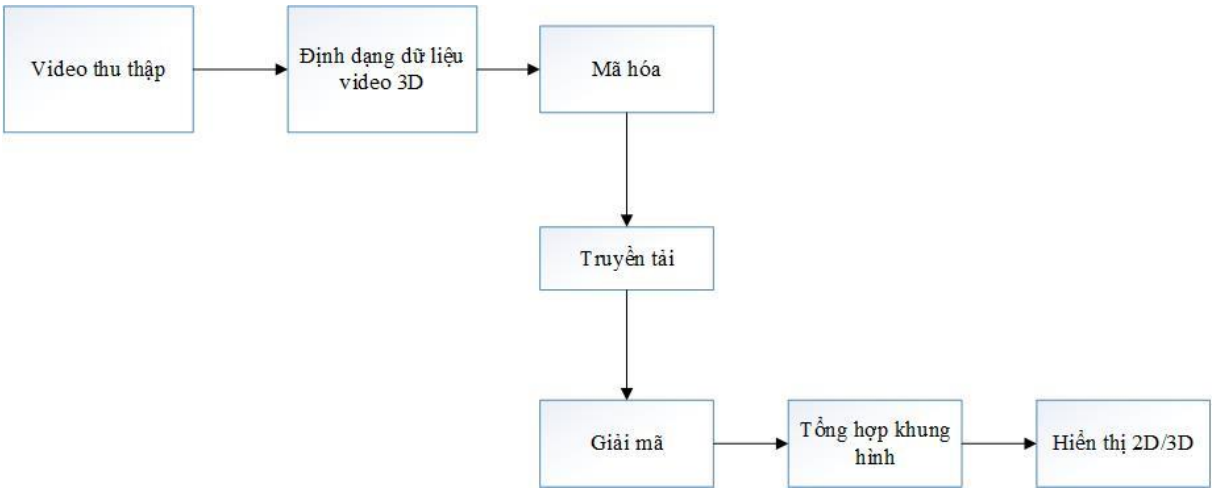


Hình 2.1: Minh họa nguyên lý nhìn của con người [8]

2.1.2. TIVI FREE VIEWPOINT (FTV)

FTV là một hệ thống cho phép người dùng kiểm soát tương tác các điểm khung hình và tạo ra các khung hình mới của một cảnh động từ bất kỳ vị trí 3D nào. FTV hứa hẹn sẽ phục vụ nhu cầu người sử dụng với mức độ cao hơn về chất lượng video. Trong một số khía cạnh, FTV là giống như đồ họa máy tính 3D, cái mà cho phép chúng ta quan sát khung cảnh từ một góc nhìn bất kỳ. Nhưng FTV có thể hiển thị những khung cảnh thực tế được chụp bởi camera thực tế trong khi đồ họa máy tính 3D chỉ có thể thực hiện hình ảnh máy tính tạo ra. FTV có thể mang lại những trải nghiệm thú vị cho người sử dụng khi áp dụng đa dạng các nội dung giải trí như là sự kiện thể thao và phim. Một hệ thống hoàn chỉnh FTV chứa đựng nhiều giai đoạn như thấy trong Hình 2.2. Trước tiên,

các cảnh được chụp bởi một hệ thống đa camera. Chúng ta cần thiết đặt camera với các đặc tính khác nhau như thể chúng là camera duy nhất. Sau đó, dữ liệu phải được mã hóa và được truyền tới người sử dụng. Ví dụ, trong cấu trúc MPEG 3DV, định dạng 3D là Video đa khung hình gồm chiều sâu (MVD) sử dụng các video 2D thông thường và thêm vào bản đồ chiều sâu với chuỗi 8 bit. Sau đó, dữ liệu phải được mã hóa và truyền tới người sử dụng. Các dữ liệu lớn vì vậy chúng ta cần phải có một chương trình nén hiệu quả. Về phía người sử dụng, dữ liệu được giải mã và sử dụng để tạo ra các khung hình mới tương thích với điểm quan sát người sử dụng. Chúng ta có thể nắm bắt được số khung hình hữu hạn để việc hiển thị khung hình tổng hợp đóng một vai trò quan trọng trong việc sản xuất nội dung cho các màn hình 3D



Hình 2.2: Hệ thống FTV tổng quát

2.2. CÁC ĐỊNH DẠNG BIỂU DIỄN VIDEO 3D

Trong kỹ thuật video, video 3D là ngày càng phổ biến bởi vì sự hữu ích của chúng trong nhiều ứng dụng. Hiển nhiên rằng, biểu diễn 3D một cách hiệu quả là cần thiết cho các ứng dụng 3D video thành công và nó cũng liên quan chặt chẽ tới các thành phần khác của hệ thống 3D video như: thu thập nội dung, truyền tải, biểu diễn và hiển thị. Hiển thị 3D linh hoạt cho cả người cung cấp lẫn người tiêu dùng sẽ có tác động đáng kể đến hiệu suất tổng thể của hệ thống, bao gồm yêu cầu về băng thông và chất lượng hình ảnh người dùng cuối cùng cũng như những hạn chế như là khả năng tương thích với các thiết bị và cơ sở hạ tầng hiện có [9]. Phần sau đây sẽ xem xét hai định dạng biểu diễn 3D: định dạng video đa khung hình (MVV) và video đa khung hình định dạng chiều sâu (MVD)

2.2.1. VIDEO ĐA KHUNG HÌNH (MVV) VÀ VIDEO ĐA KHUNG HÌNH THEO CHIỀU SÂU (MVVD)

Video đa khung hình (MVV) là một định dạng video bao gồm một vài video màu từ các điểm khung hình khác nhau của cùng một cảnh đạt được bởi 1 hệ thống camera như Hình 2.3 . MVV đặc biệt là thích hợp cho hiển thị tự động lập thể, yêu cầu một lượng lớn khung hình. Hơn nữa, nó cũng cho phép lưu giữ toàn bộ độ phân giải của chuỗi video [9]. Ngoài ra những khó khăn liên quan đến tổng hợp khung hình có thể tránh được. Cuối cùng, việc hiển thị có thể dễ dàng được thực hiện tương ứng với hiển thị 2D truyền thống bằng cách trích xuất từ 1 trong các khung hình. Tùy thuộc vào mục đích cụ thể, số lượng camera và sự sắp xếp camera có thể khác nhau. Thông thường, có 3 kiểu sắp xếp camera: sắp xếp tuyến tính, sắp xếp phẳng và sắp xếp hình tròn như Hình 2.4

Video đa khung hình (MVV) là 1 định dạng video bao gồm một vài video màu từ các điểm khung hình khác nhau trong cùng một cảnh được đồng bộ bởi một hệ thống camera được hiển thị như Hình 2.3. MVV đặc biệt thích hợp cho màn hình lập thể tự động, những màn hình này yêu cầu số lượng lớn các khung hình. Hơn thế nữa, màn hình này cho phép bảo toàn được toàn bộ độ phân giải chuỗi video. Ngoài ra, những khó khăn liên quan đến tổng hợp khung hình có thể tránh được. Cuối cùng, việc hiện thị có thể dễ dàng được thực hiện tương thích với các màn hình truyền thống 2D bằng cách trích xuất ra 1 trong các khung hình. Tùy thuộc vào các mục đích cụ thể, số lượng camera và sự sắp xếp các camera có thể khác nhau. Thông thường, có 3 kiểu bố trí camera: tuyến tính, phẳng và tròn như Hình 2.4

Hình 2.3: Ví dụ về một cảnh biểu diễn video đa khung hình – Break Dance

Hình 2.4: Ví dụ về sắp xếp một hệ thống camera đa khung hình

Mã hóa video đa khung hình có thể được nén một cách hiệu quả nội dung MVV bằng cách kết hợp dự đoán dựa trên chuyển động trong khung hình thông thường và dự đoán dựa trên độ lệch trong khung hình nhưng tỉ lệ bit vẫn tăng lên một cách tuyến tính với số lượng khung hình được mã hóa. Điều này dẫn đến sự xuất hiện định dạng chiều sâu với video đa khung hình (MVD). MVD là 1 sự kết hợp của MVV và định dạng chiều sâu với video. Vì vậy, nó có những lợi thế từ cả hai. Trong MVD, mỗi khung hình thứ N được yêu cầu với chiều sâu liên quan, như Hình 2.5 . Với thông tin chiều sâu từ mỗi khung hình, MVD chứng minh rằng hình học 3D của cảnh với độ chính xác tốt hơn nhiều so với MVV hoặc video theo chiều sâu. Vì vậy, chúng ta có thể áp dụng kỹ thuật biểu diễn hình ảnh DIBR để biểu diễn các khung hình trung gian tại bất kỳ vị trí cuối nào của người nhận. Điều này giúp giảm số lượng khung hình cần để truyền tải so với trường hợp MVV. Do đó, MVD là một trong những định dạng phổ biến nhất để hiển thị video 3D. Hai chuỗi, vân video và độ sâu có thể được mã hóa và được truyền đi một cách độc lập hoặc có thể cùng được mã hóa bằng việc khai thác các dư thừa giữa chúng để đạt được hiệu suất mã hóa tốt hơn

Hình 2.5: Ví dụ về video đa khung hình với chiều sâu

2.2.2. BẢN ĐỒ ĐỘ SÂU

Bản đồ chiều sâu (ảnh chiều sâu) là một ảnh với kích thước bằng với ảnh màu, giá trị của mỗi điểm ảnh trong ảnh chiều sâu là giá trị chiều sâu của điểm ảnh màu tương ứng, như được chỉ thấy trong Hình 2.6 . Nói cách khác, một bản đồ chiều sâu ánh xạ mỗi điểm ảnh trong một video màu để khoảng cách của nó từ camera (trục Z trên camera). Bản đồ độ sâu chủ yếu bao gồm các vùng mịn được ngăn cách bởi các biên mà không có vân hay bóng. Điển hình bản đồ độ sâu là một ảnh gray scale 8 bit, khoảng giá trị bit từ 0 đến 255. Giá trị 0 là giá trị ở gần mặt phẳng nhất (Z_{near}) biểu diễn mức xa nhất và giá trị 255 là giá trị cách xa mặt phẳng nhất (Z_{far}) biểu diễn mức độ gần nhất

Hình 2.6: Một khung màu và bản đồ độ sâu liên quan

Có hai hướng tiếp cận để xây dựng bản đồ chiều sâu. Hướng tiếp cận thứ nhất được tích hợp vào một camera thời gian bay (ToF) [10] để tính toán khoảng cách từ các điểm trong khung cảnh đến camera. Camera ToF là một hệ thống camera sắp xếp để giải quyết khoảng cách dựa vào tốc độ ánh sáng, đo lường thời gian bay của một tín hiệu ánh sáng giữa camera và đối tượng của mỗi điểm trên ảnh. Kỹ thuật này mang lại các kết

quả hữu ích nhưng nó chỉ có hiệu quả bên trong một vùng nhỏ với độ sâu lên đến vài mét. Một hướng tiếp cận khác dựa trên sự có sẵn của các quan sát khác nhau của cùng hình ảnh. Bằng cách so sánh hai hình ảnh của cùng một khung cảnh, thông tin về chiều sâu có thể đạt được trong hình thái của một bản đồ độ lệch được mã hóa khác biệt hệ tọa độ của các điểm ảnh tương ứng [12]. Khái niệm độ lệch được minh họa rõ ràng trong Hình 2.7. Các giá trị trong bản đồ độ lệch là tỉ lệ nghịch với độ sâu khung cảnh ở mỗi vị trí điểm ảnh tương ứng. Độ lệch của một đối tượng trong hệ tọa độ camera có thể được suy ra một cách dễ dàng từ công thức sau:

$$\boldsymbol{disparity = x_l - x_r = \frac{f * T}{Z * t_{pixel}}} \qquad \textbf{(1)}$$

ở đây:

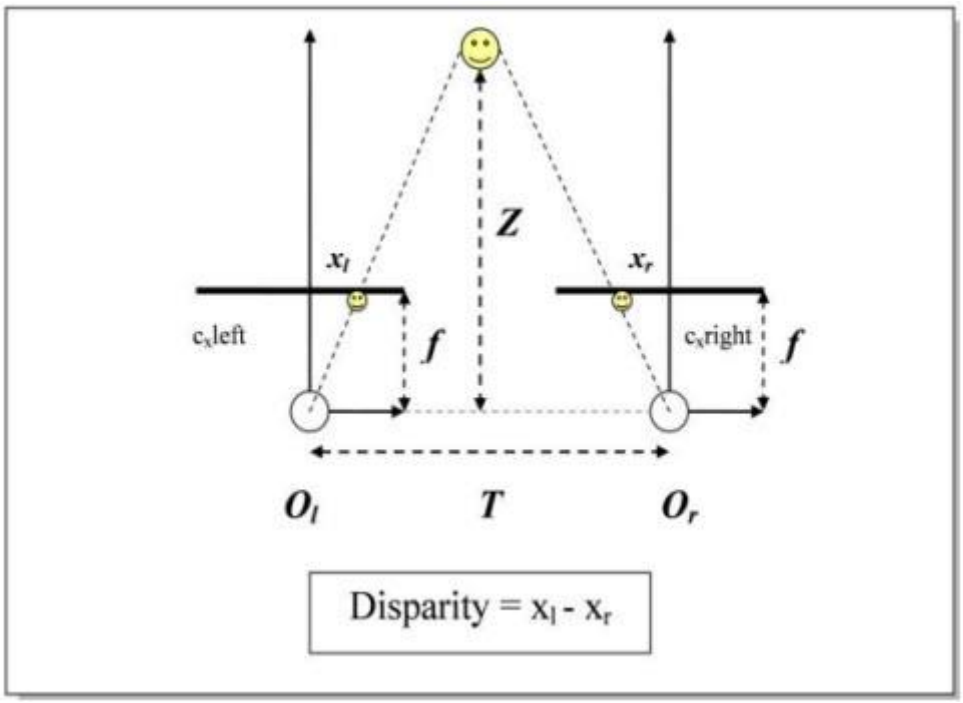
$\boldsymbol{x_l}$ và $\boldsymbol{x_r}$ là vị trí của đối tượng bên trái và bên phải camera tương ứng.

f là chiều dài tiêu cự. T là khoảng cách giữa camera (cơ bản).

Z là khoảng cách giữa đối tượng và mặt phẳng ảnh của camera chụp.

t_{pixel} là độ rộng của một điểm ảnh trên cảm biến camera.

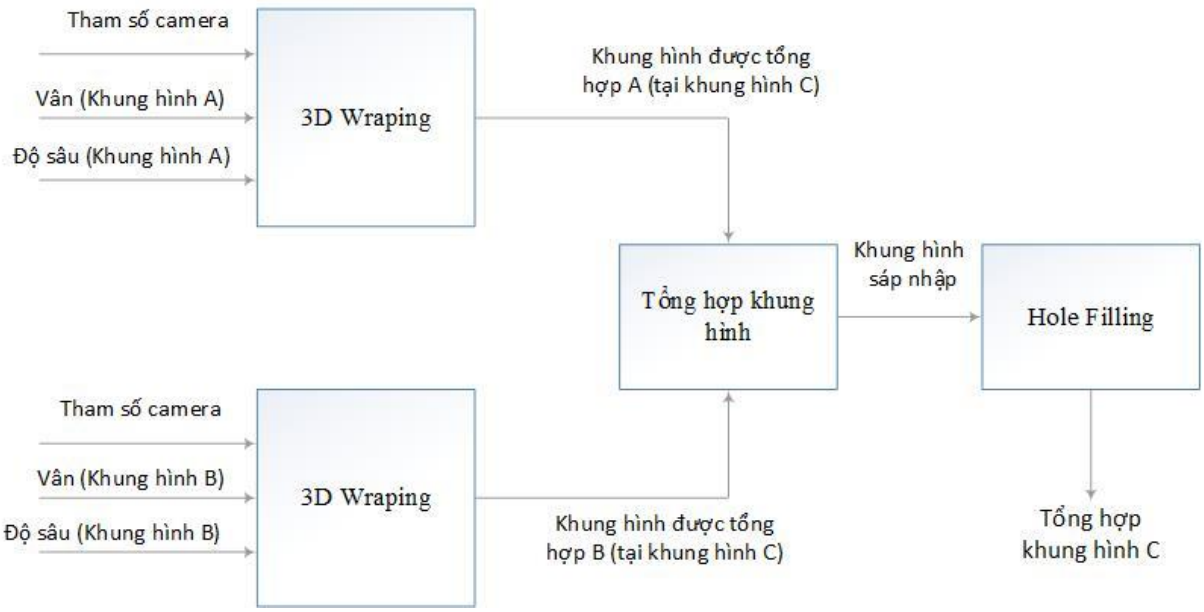
Bằng cách này, vấn đề ước lượng chiều sâu được chuyển thành việc tính toán sự chênh lệch hoặc tìm điểm ảnh tương ứng trong ảnh. Bởi vì tính hữu dụng và giá trị của khái niệm này , phần mềm DERS (Depth Estimation Reference Software) [11] được phát triển bởi MPEG là một phần mềm tham chiếu cho việc ước lượng bản đồ độ sâu từ chuỗi các hình ảnh được chụp bởi một tập hợp nhiều camera.



Hình 2.7: Công thức tính độ lệch

2.3. BIỂU DIỄN DỰA TRÊN BẢN ĐỒ ĐỘ SÂU (DIBR)

Biểu diễn dựa trên độ sâu ảnh (Depth-Image-Based Rendering - DIBR) [4] là quá trình tổng hợp ảnh các khung hình ảo từ cảnh được chụp từ ảnh hoặc video màu với thông tin độ sâu liên quan [13]. Với M (M ≥ 1) các khung hình đầu vào (còn gọi là khung hình tham chiếu) , một khung hình ảo có thể được tổng hợp thông qua ba bước chính sau. Trước tiên, các điểm ảnh trong khung hình tham chiếu có thể được chiếu đến khung hình ảo đích, quá trình này gọi là 3D wraping. Tiếp theo, các điểm ảnh từ các khung hình tham chiếu được chiếu đến vị trí giống nhau trong khung hình ảo, quá trình này được gọi là view merging. Cuối cùng, còn lại các hố (các vị trí mà không có điểm ảnh nào được chiếu) trong khung hình ảo được lấp đầy bằng cách tạo ra các thành phần vân trực quan phù hợp với các điểm ảnh lân cận, quá trình này gọi là hole filling. Các bước này được minh họa trong Hình 2.8 và sẽ được miêu tả rõ ràng hơn trong các phần dưới đây



Hình 2.8: Framework khung hình tổng hợp cơ bản sử dụng 2 camera đầu vào

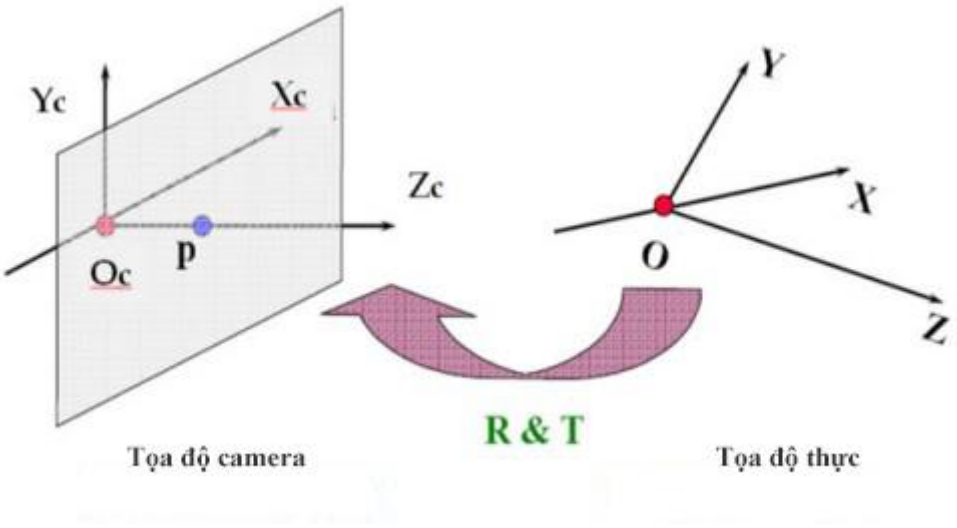
2.3.1. TỔNG HỢP KHUNG HÌNH 3D TỪ 2D

Tổng hợp 3D (3D Wraping) được sử dụng để xác định tọa độ thực của một hình ảnh có sử dụng các thông số bên trong và bên ngoài máy ảnh. Sau đó, tổng hợp 3D được sử dụng để tạo ra hình ảnh mong muốn thông qua việc tái chiếu không gian 2D sử dụng các tham số camera ảo. Việc chuyển đổi hệ thống hình ảnh 2D thành hệ thống hình ảnh 3D là điều cần thiết cho quá trình tổng hợp 3D và hệ thống thế giới thực 3D, hệ thống thế giới thực 2D được đưa ra cần được chuyển đổi thành hệ tọa độ camera 3D. Hệ tọa độ thực và các hệ tọa độ camera cả 2 là các hệ tọa độ 3D và việc chuyển đổi giữa hai hệ thống có thể đạt được thông qua việc quay và dịch chuyển như Hình 2.9. Hai hệ thống này được định nghĩa như là tham số bên ngoài camera

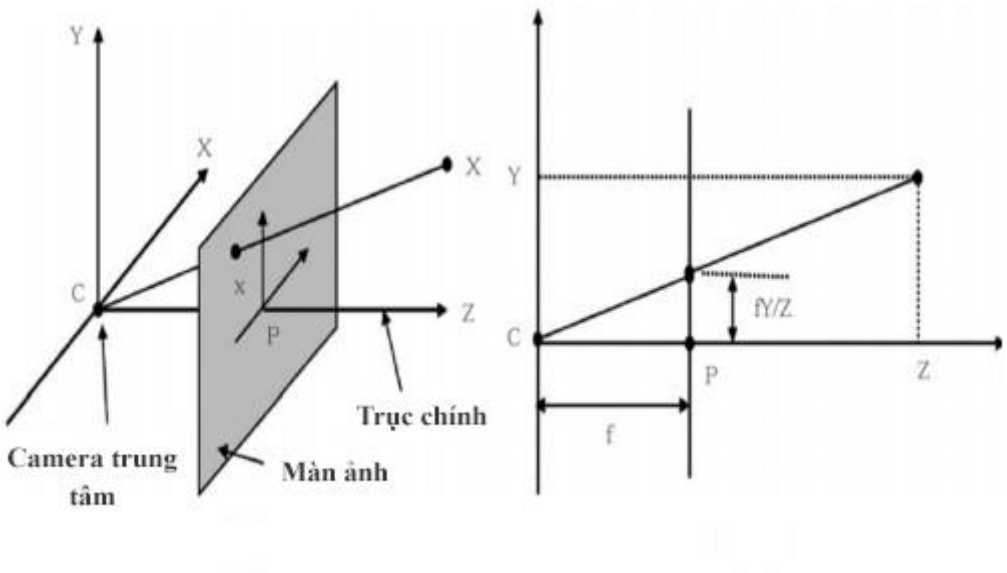
Việc chuyển đổi hệ tọa độ camera thành hệ tọa độ hình ảnh 2D có thể được giải thích thông qua cấu trúc hình học của camera như được chỉ ra trong Hình 2.9. Hình 2.10 (a) giới thiệu mô hình 3D của camera pin-hole và Hình 2.10 (b) giới thiệu mô hình 2D. Nhìn chung, ảnh từ camera pin-hole đi qua lỗ theo một đường thẳng và hình thành một hình ngược ở vị trí f của trục Z (f là tiêu cự của máy ảnh). Tuy nhiên, vách ngăn nơi một bức ảnh hình thành được di chuyển đến chiều dài tiêu cự trên trục Z sau khi phân tích

Chiếu tọa độ 3D của một đối tượng của hệ tọa độ camera trên màn hình ảnh có thể được giải thích bằng hình tam giác được hình thành sử dụng độ dài tiêu cự và tọa độ của đối tượng như được chỉ trong Hình 2.10. Quá trình chuyển đổi được định nghĩa như

là tham số nội tại . Sử dụng các thông số bên trong và bên ngoài của camera, hệ tọa độ trong hệ thống tọa độ thế giới thực có thể được chuyển đổi thành hệ tọa độ 2D trong màn hình ảnh như được chỉ ra trong công thức 2



Hình 2.9: Chuyển đổi hệ tọa độ thực sang hệ tọa độ camera



(a) **(b)**

Hình 2.10: Cấu trúc hình học của camera pin-hole (a) 3D và (b) 2D

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|T] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \qquad (\mathbf{2})$$

Trong đó: x,y là tọa độ 2D của màn ảnh

K là tham số bên trong của camera

R là ma trận xoay của camera

T là vector dịch của camera

X,Y,Z là tọa độ 3D của hệ tọa độ thực

$K[R|T]$ là ma trận chiếu

Thông qua ma trận nghịch đảo trong công thức 2, tọa độ 2D có thể được chuyển thành hệ tọa độ thực. Lúc này, thông tin độ lệch D từ công thức 3 cần tìm giá trị độ sâu thực Z

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = K[R|T] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} => K^{-1} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = R \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} + T$$

$$=> R^T K^{-1} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} - R^T T = \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$Z_{(i,j)} = \frac{1.0}{\left(\frac{D_{(i,j)}}{255.0} \times \left(\frac{1.0}{MinZ} - \frac{1.0}{MaxZ}\right) + \frac{1.0}{MaxZ}\right)} \qquad (\mathbf{3})$$

Trong đó: $Z_{(i,j)}$ và $D_{(i,j)}$ là giá trị độ sâu và giá trị độ lệch của tọa độ (i, j) trong ảnh

MinZ và MaxZ là giá trị nhỏ nhất và lớn nhất của Z tương ứng

Để tạo ra ảnh ảo, các tham số bên trong và bên ngoài của camera ảo tồn tại ở một vị trí ảo cần được xác định. Nhìn chung, tham số bên trong được xác định bởi cấu trúc bên trong của camera. Do đó các tham số bên trong của điểm tham chiếu camera có thể được sử dụng như tham số bên trong của camera ảo

(a) **(b)**

(c) **(d)**

(e) **(f)**

Hình 2.11: Tổng hợp khung hình với hai khung hình dữ liệu MVD (a) Vân ảnh của khung hình tham chiếu trái (b) Bản đồ độ sâu liên quan của khung hình tham chiếu trái. (c) Khung hình trái được tổng hợp. (d) Khung hình phải được tổng hợp . (e) Khung hình pha trộn với các hố còn lại. (f) Khung hình tổng hợp cuối cùng sau khi loại bỏ hố trống

Giá trị của tham số bên trong được sử dụng sau khi chuyển vị trí của camera ảo. Hệ tọa độ thực 3D được chuyển đổi và thông số của camera ảo được áp dụng cho công thức (2) để tìm ra công thức (4). Sau đó, công thức (4) biểu thị lại tọa độ ảnh của điểm ảo

$$\begin{bmatrix} x_v \\ y_v \\ 1 \end{bmatrix} = K_V [R_V | T_V] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \qquad (4)$$

Ở đây: x_v, y_v biểu diễn tọa độ 2D của ảnh ảo được hình thành

K_V, R_V, T_V biểu diễn tham số bên trong, ma trận xoay và vector dịch chuyển của camera ảo tương ứng

Hình 2.11 chỉ ra ví dụ một điểm ảnh ảo được sinh ra bởi kỹ thuật tổng hợp 3D. Như trong hình , trong ảnh ảo được hình thành, một vùng occlusion có thể được tìm thấy ở ảnh tham chiếu, Vùng này gọi là vùng common-hole (hố chung)

2.3.2. SÁP NHẬP KHUNG HÌNH

Tổng hợp khung hình có thể được phân thành hai phương pháp. Phương pháp thứ nhất là nội suy khung hình có nghĩa là khung hình ảo (đích) nằm trong hai khung hình tham chiếu tồn tại, ở đây thông tin màu sắc và chiều sâu từ cả hai khung hình có thể được sử dụng để tạo ra khung hình trung gian. Phương pháp thứ 2 là ngoại suy khung hình có nghĩa là khung hình ảo nằm ngoài các khung hình tồn tại. Trong phương pháp ngoại suy khung hình, chỉ có thông tin về màu và chiều sâu từ một khung hình đơn có thể được sử dụng cho quá trình biểu diễn. Để đạt được kết quả tốt nhất, khung hình ảo phải được thay thế giữa hai khung hình tham chiếu bởi vì, trong trường hợp này, các hố trong một khung hình được tổng hợp thường thì có thể được bổ sung bằng các vùng không có hố tương ứng trong khung hình tổng hợp khác. Sau khi chồng tất cả các khung hình được tổng hợp và sáp nhập chúng thành một ảnh, các hố được loại bỏ một cách đáng kể. Có một vài cách để kết hợp các khung hình được giới thiệu. Phương pháp thứ nhất là trọng số trung bình là sư pha trộn các điểm ảnh có sẵn từ hai khung hình được tổng hợp với một hàm trọng số tuyến tính

$$V_i = \frac{|t_{x,L-V}| R_i + |t_{x,R-V}| L_i}{|t_{x,L-V}| + |t_{x,R-V}|} \qquad (5)$$

Ở đây V_i, L_i, R_i biểu thị điểm ảnh thứ i_{th} trong khung hình ảo, khung hình tham chiếu bên phải và khung hình tham chiếu bên trái một cách tương ứng; $|t_{x,L-V}|$ là khoảng cách cơ bản giữa khung hình bên trái với khung hình ảo. Thật rõ để thấy rằng pha trộn hai khung hình được tổng hợp bằng cách đưa ra trọng số lớn hơn đối với khung hình tham chiếu là gần hơn đối với khung hình ảo.Và nếu có những hình giả (artifacts) trong hai khung hình tổng hợp, chúng vẫn sẽ được hiển thị trong khung hình được sáp nhập, mặc dù chúng được giảm trừ đi do trọng số pha trộn

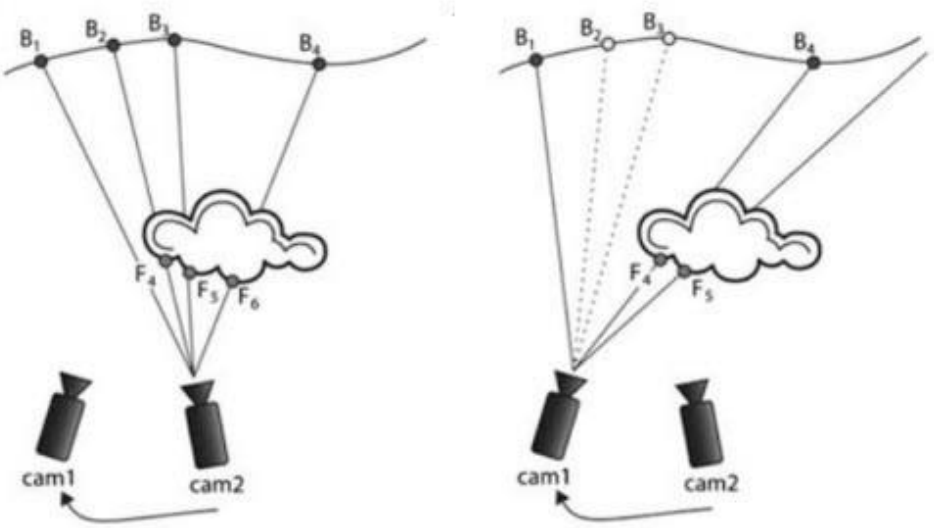
Phương án thứ hai được chọn hoặc là từ khung hình được tổng hợp bên trái hoặc từ khung hình tổng hợp bên phải được gọi là khung hình chi phối và các điểm ảnh từ khung hình được tổng hợp khác chỉ được sử dụng để lấp các hố trong khung hình chi phối. Trong trường hợp một khung hình được tổng hợp tốt hơn các khung hình khác, chất lượng của khung hình tổng hợp có thể cao hơn khi so sánh với phương pháp trọng số trung bình. Ngoài ra, khung hình được kết hợp cũng có độ tương phản hơn so với trong số trung bình bởi vì pha trộn hai khung hình được tổng hợp dễ dàng dẫn đến hiệu ứng làm mờ khi các thành phần vân từ hai khung hình được tổng hợp không được căn chỉnh tốt

Phương pháp cuối cùng là lựa chọn điểm ảnh với giá trị độ sâu cao hơn dựa trên phương pháp z-buffer [14]. Phương pháp này làm việc tốt khi bản đồ độ sâu không có lỗi. Tuy nhiên, phương pháp này có xu hướng tạo ra các hình giả khi dữ liệu độ sâu tạm thời không phù hợp. Các giá trị độ sâu khác nhau có thể dẫn đến trường hợp một điểm ảnh lấy giá trị từ hai khung hình tham chiếu tương ứng tại hai thời điểm liên tiếp. Thông thường, các giá trị điểm ảnh như nhau từ hai khung hình có thể khác nhau về màu sắc để phiên của hai khung hình có thể là nguyên nhân gây ra sự thống nhất không tạm thời trong khung hình ảo

2.3.3. HOLE FILLING CÁC VÙNG DISOCCLUSIONS

Vấn đề chính của DIBR là một vài điểm ảnh trong khung hình ảo không tồn tại trong khung hình tham chiếu và ngược lại được chỉ ra trong Hình 2.12 . Hai điểm B_2 và B_3 có thể được nhìn thấy trong $cam1$ nhưng không nhìn thấy trong $cam2$. Mặt khác, các vùng nhận định không nhìn thấy trong các khung hình camera gốc trở thành nhìn thấy trong khung hình đích. Những vùng disocclusion được gọi là các hole. Đặc biệt,

trong trường hợp ngoại suy khung hình, ở đây các khung hình đích nằm ngoài các đường camera cơ sở đang tồn tại, các hố xuất hiện bởi vì không có thông tin những vùng trong các khung hình tham chiếu.



(a) **(b)**

Hình 2.12: Cấu hình lập thể, tất cả điểm ảnh không nhìn thấy từ các điểm quan sát camera

Để cung cấp cho người xem trải nghiệm hoàn thiện, các hố trong khung hình biểu diễn cần được loại bỏ. Có hai hướng chính để giải quyết vấn đề này. Một hướng là xử lý trước bản đồ độ sâu bằng cách làm mịn vùng không liên tục của bản đồ độ sâu trước khi dùng phương pháp DIBR loại bỏ vùng disocclusion trong khung hình tổng hợp. Phương pháp này nhằm giải quyết vấn đề lấp đầy trong các vùng disocclusion trong trường hợp khoảng cách camera nhỏ [15]. Hướng tiếp cận khác là xử lý sau khung hình tổng hợp để lấp đầy các vùng còn thiếu thông tin. Quá trình này được gọi là “holes filling”. Phương pháp này có thể áp dụng đối với một thiết lập camera có đường cơ sở lớn. Một vài kỹ thuật hole filling được đề xuất với mức độ phức tạp khác nhau cũng như sự cải thiện về chất lượng. Các kỹ thuật hole filling khác nhau từ việc sao chép một điểm ảnh liên tục đến phương pháp inpainting phức tạp [16,17]. Các ví dụ của các phương pháp hole filing được chỉ ra trong Hình 2.13

Hình 2.13: Phương pháp hole filling truyền thống

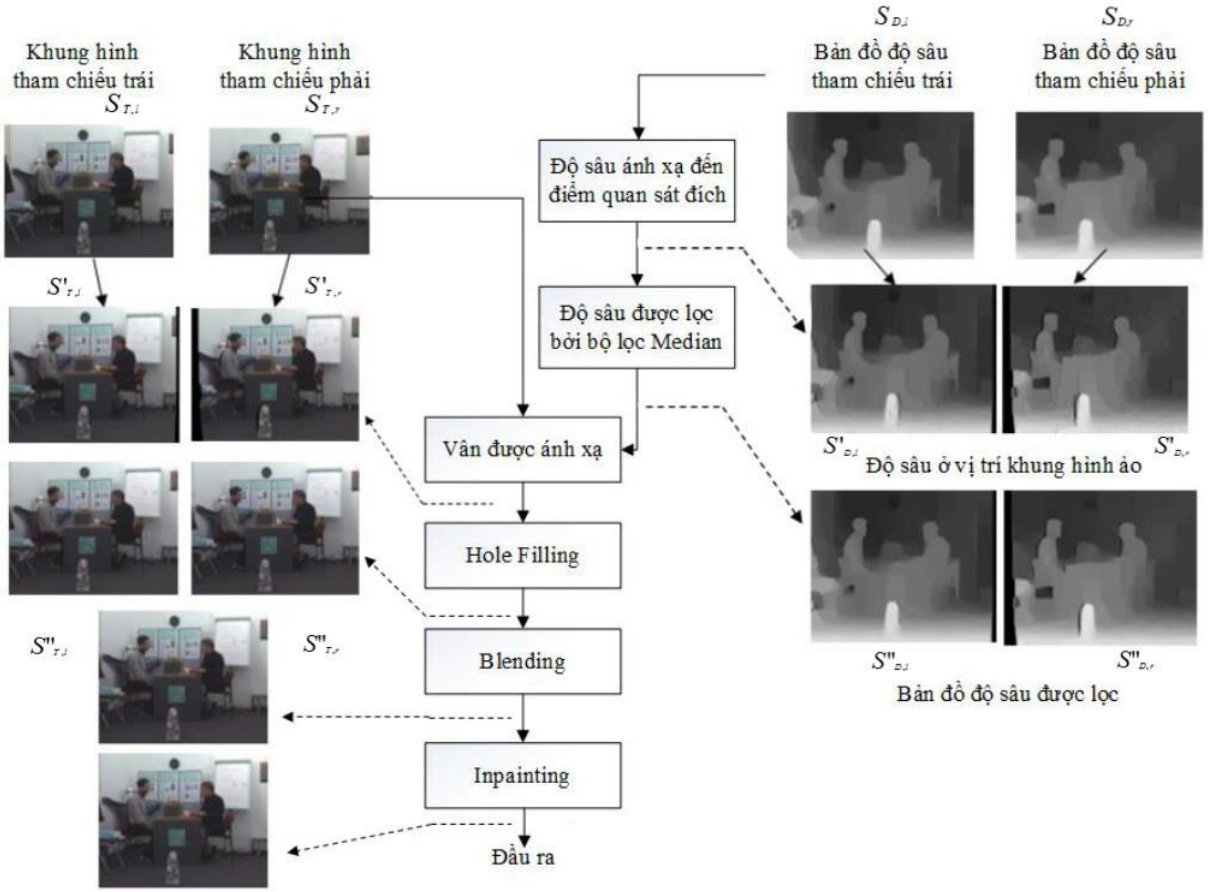
2.4. PHẦN MỀM THAM CHIẾU TỔNG HỢP KHUNG HÌNH (VSRS)

Phần mềm tham chiếu tổng hợp khung hình (VSRS) được phát triển như là một phần của các thí nghiệm nghiên cứu 3DV của MPEG. VSRS cần hai khung hình tham chiếu và hai bản đồ độ sâu là đầu vào để tạo ra một khung hình ảo tổng hợp. Các tham số bên trong và bên ngoài camera được yêu cầu và các thiết đặt camera 1D song song và không song song được hỗ trợ. Phần mềm tham chiếu có hai chế độ chính được gọi là "trạng thái tổng quát" và "trạng thái 1D". Các khung hình tham chiếu được ánh xạ lại đến các điểm khung hình đích có sử dụng ánh xạ từng điểm ảnh với nhau dựa trên tổng hợp 3D trong "trạng thái tổng quát" hoặc chuyển đổi điểm ảnh theo chiều ngang trong "trạng thái 1D". Các chi tiết về chúng được miêu tả trong các phần sau

2.4.1. TRẠNG THÁI TỔNG QUÁT

Quá trình biểu diễn trong trạng thái tổng quát trong VSRS được minh họa trong Hình 2.14. Trước tiên, các bản đồ độ sâu tham chiếu bên trái và bên phải ($S_{D,l}$) và ($S_{D,r}$) được tổng hợp thành khung hình ảo, sinh ra ($S'_{D,l}$) và ($S'_{D,r}$) sử dụng kỹ thuật tổng hợp 3D được mô tả chi tiết như trong mục 2.3.1. Nếu có nhiều điểm ảnh trong khung hình

tham chiếu được tổng hợp đến vị trí giống nhau trong khung hình ảo thì điểm ảnh đó có giá trị độ sâu cao nhất (gần camera) được sử dụng. Mặt khác, tiền cảnh sẽ hấp thụ nền. Bản đồ độ sâu được tổng hợp ($S'_{D,l}$) và ($S'_{D,r}$) có thể có những hố nhỏ hơn bởi vì vị trí của các điểm ảnh tổng hợp được làm tròn từ số thực sang số nguyên. Bộ lọc Median được áp dụng để lấp đầy các loại hố, sinh ra trong ($S''_{D,l}$) và ($S''_{D,r}$). Ngoài ra, một mặt nạ nhị phân được duy trì cho mỗi bên khung hình tham chiếu để đánh dấu là các hố do các occlusions sinh ra và vẫn còn sau khi lọc



Hình 2.14: Biểu đồ luồng dữ liệu của phần mềm VSRS trạng thái tổng quát

Sau đó, $S''_{D,l}$ và $S''_{D,r}$ được sử dụng để tổng hợp các khung hình tham chiếu vân bên trái và bên phải ($S_{T,l}$) và ($S_{T,r}$) đến vị trí khung hình ảo. Vì vậy, hai ảnh vân được tổng hợp thu được ($S'_{T,l}$) và ($S'_{T,r}$) một ảnh từ khung hình trái và một ảnh khác từ khung hình phải. Các hố gây ra bởi occlusion trong một ảnh vân được tổng hợp được lấp đầy bởi các điểm ảnh không có hố từ các vùng vân khác nếu có sẵn. Vì vậy, các hố trong $S'_{T,l}$ được lấp đầy bởi các vùng không có hố trong $S'_{T,r}$ và ngược lại. Sau bước này, chúng ta có hai khung hình ảo được lấp là $S''_{T,l}$ và $S''_{T,r}$ được sáp nhập thành một

khung hình ảo duy nhất. Trạng thái tổng quát có hai tùy chọn kết hợp: trạng thái trộn bật và trạng thái trộn tắt (Blending-on và Blending-off). Trạng thái trộn bật liên quan đến trọng số trung bình dựa trên khoảng cách mỗi khung hình tham chiếu đến khung hình ảo, những điểm ảnh từ khung hình tham chiếu gần hơn sẽ có trọng số cao hơn các khung hình khác. Trong trạng thái Blending-off, việc đơn giản là lấy các giá trị điểm ảnh từ khung hình tham chiếu gần hơn và bỏ các giá trị điểm ảnh từ các khung hình tham chiếu khác. Các hố chỉ được lấp đầy từ các khung hình tham chiếu khác. Các mặt nạ nhị phân của mỗi khung hình cũng được sáp nhập bởi một kĩ thuật ảnh inpainting dựa trên phương pháp tiến hành nhanh. Trong bước inpainting, thông tin màu sắc được lan truyền từ bên trong biên các hố. Ngoài ra, VSRS bao gồm một tùy chọn gọi là Boundary Noise Removal

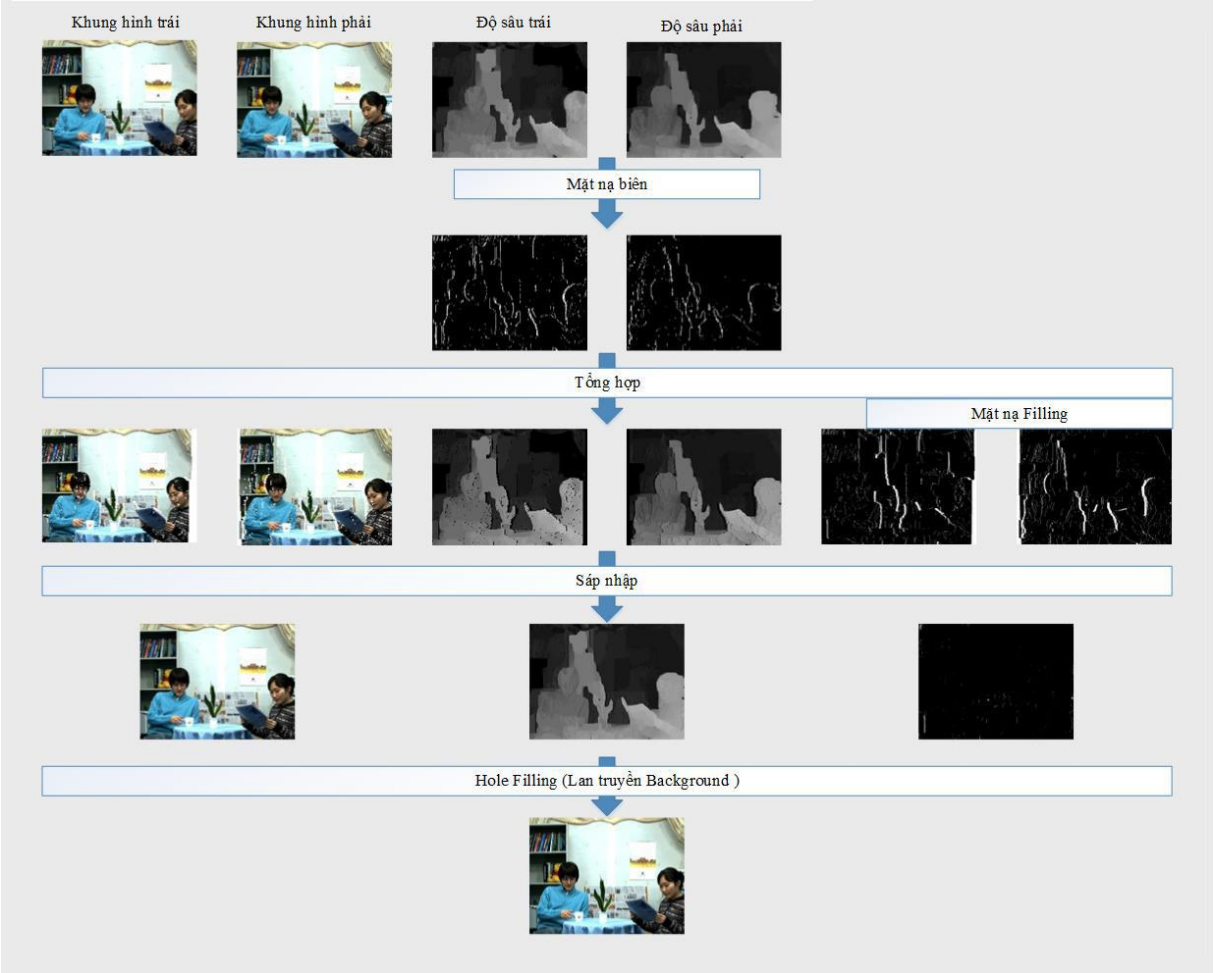
Trong trạng thái này, phần nền của các hố được xác định dựa trên giá trị độ sâu và các hố được mở rộng trên phần nền. Sau đó, những vùng này trong $S'_{T,l}$ và $S'_{T,r}$ được lấp đầy từ khung hình tham chiếu đối diện. Điều đó có thể làm giảm các trường hợp mà ở đây các điểm ảnh trước nền được ánh xạ sai vào nền đối tượng do các lỗi về độ sâu

2.4.2. TRẠNG THÁI 1D

Trong trạng thái 1D của VSRS, có một giả định rằng các trục quang học của máy ảnh là song song và không có độ lệch nào theo phương thẳng đứng. Sự thiệt đặt này làm cho tiến trình tổng hợp 3D giảm đi sự thay đổi đơn giản theo chiều ngang và được chi tiết trong công thức (9). Hình 21 mô tả biểu đồ luồng của VSRS 1D-mode. Tại bước đầu tiên, các thành phần Chroma được up-sample với định dạng 4:4:4 (để thực hiện đơn giản), bản đồ độ sâu có thể được lọc bỏ tạm thời để giảm lỗi độ sâu và video màu được up-sample với độ chính xác điểm ảnh ½ hoặc ¼ . Các bản đồ độ sâu và các khung hình tham chiếu được tổng hợp thành khung hình ảo sử dụng công thức (9). Nhìn chung với trạng thái tổng quát, cho mỗi khung hình tham chiếu, một mặt nạ nhị phân (mặt nạ filling) được duy trì để đánh dấu nếu một điểm ảnh được lấp đầy hoặc không lấp đầy hố. Ngoài ra, có hai tiến trình tăng cường là : “CleanNoiseOption” và“WarpEnhancementOption” để loại bỏ các hình giả tổng hợp do sự sai lệch giữa vân và độ sâu ở biên của đối tượng (điều đó làm cho các điểm ảnh nền tổng hợp đến nền). Tiếp theo, hai vân ảnh được tổng hợp được sáp nhập hình thành nên một. Một tiến trình tương tự được áp dụng cho các bản đồ độ sâu và các mặt nạ filling. Khi một điểm ảnh

có sẵn từ cả hai khung hình tham chiếu, quá trình pha trộn được xác định trước được sử dụng.

Có 3 phương pháp pha trộn trong trạng thái này: sử dụng các điểm ảnh từ khung hình tham chiếu gần hơn so với khung hình ảo, trọng số trung bình dựa trên khoảng cách mỗi khung hình tham chiếu so với khung hình ảo và quá trình pha trộn dựa trên sự khác nhau về độ sâu. Sau cùng, bước hole filling được thực hiện bằng cách lan truyền các điểm ảnh nền vào trong hố dọc theo hàng ngang. Cuối cùng ảnh khung hình được down-sample đến kích thước gốc nếu cần thiết và chuyển thành định dạng 4:2:0 cho mục đích đầu ra

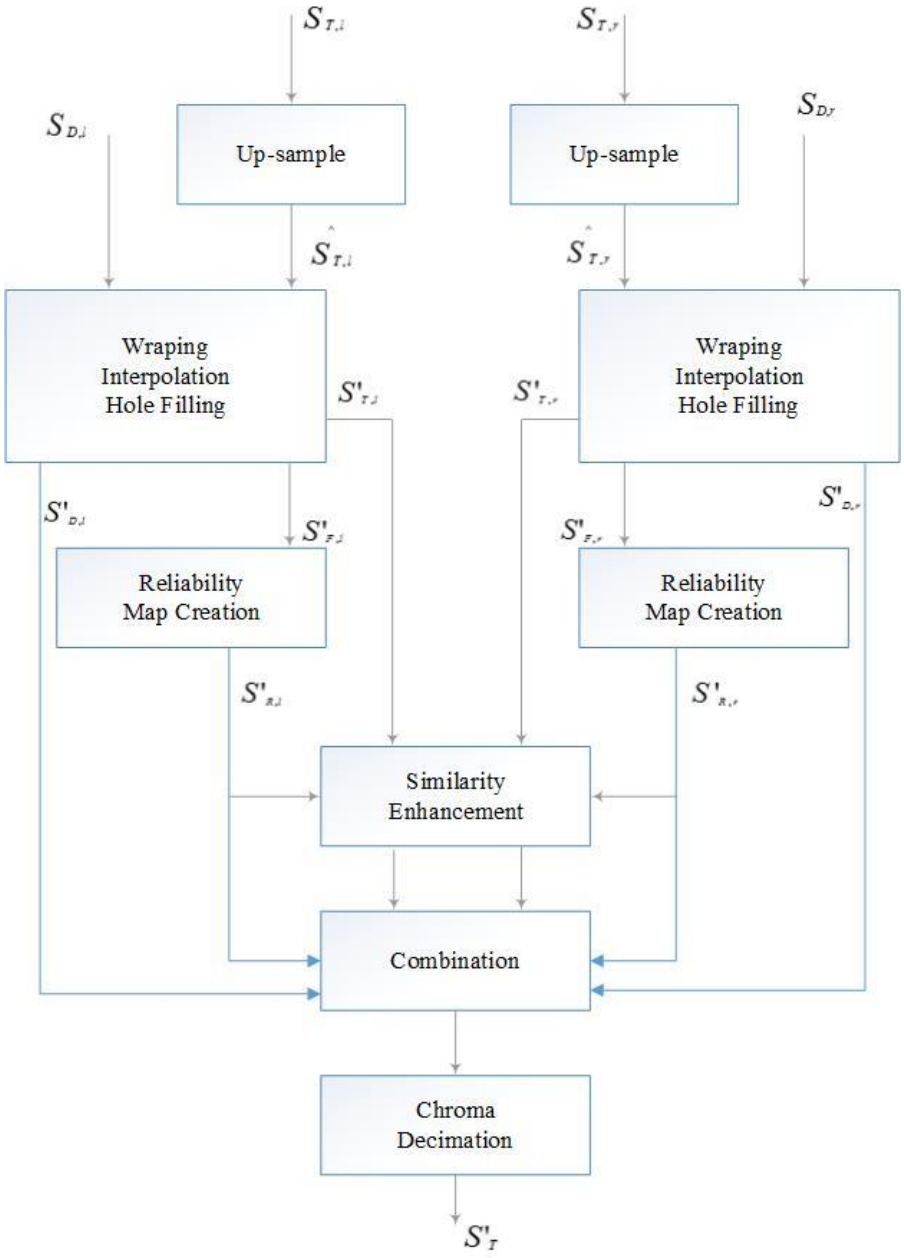


Hình 2.15: Biểu đồ luồng phần mềm VSRS 1D mode

2.5. THUẬT TOÁN TỔNG HỢP KHUNG HÌNH FAST 1-D (VSRS 1D FAST)

VSRS 1D Fast là một biến thể của VSRS. Nó được phát triển theo chuẩn 3D-HEVC để kiểm thử các kết quả mã hóa trên các khung hình tổng hợp. Phần mềm này bao gồm trong gói HTM và được dẫn chứng trong kiểu kiểm thử 3D-HEVC. VSRS 1D

Fast cần 2 hoặc 3 khung hình tham chiếu, các bản đồ độ sâu và các tham số camera tương ứng như là kết quả đầu vào để sinh ra một khung hình ảo. VSRS 1D Fast cũng yêu cầu thiết đặt camera là trục song song 1D. Có hai cấu hình trong VSRS 1D Fast: trạng thái nội suy tổng hợp khung hình ảo sử dụng cả khung hình ảo và trạng thái nội suy biểu diễn khung hình ảo một cách chính thức từ một khung hình tham chiếu; các khung hình khác được sử dụng cho hole filling. Tổng quan phần mềm VSRS 1D Fast được minh họa trong Hình 2.15. Hai vân $S_{T,l}$ và $S_{T,r}$ cùng với hai bản đồ độ sâu tương ứng $S_{D,l}$ và $S_{D,r}$ được sử dụng để tổng hợp thành khung hình ảo S'_T



Hình 2.16 : Thuật toán tổng hợp khung hình

Một cái nhìn tổng quát về phương pháp tổng hợp khung hình được mô tả như hình trên. Phương pháp này hỗ trợ nội suy của một khung hình tổng hợp từ một vân trái $s_{T,l}$ và một vân phải $s_{T,r}$ với bản đồ độ sâu tương ứng $s_{D,l}$ và $s_{D,r}$. Với phương pháp này, hai vân $s'_{T,l}$ và $s'_{T,r}$ được ngoại suy từ khung hình trái và phải ở vị trí của khung hình ảo. Sau đó, sự giống nhau giữa $s'_{T,l}$ và $s'_{T,r}$ được tăng cường trước khi kết hợp chúng thành khung hình tổng hợp đầu ra s'_T. Các bước xử lý đơn được thảo luận dưới đây. Không mất đi các bước tổng quát được thực hiện một cách độc lập cho cả hai khung hình, khung hình trái và khung hình phải, ta chỉ xét khung hình trái

Một cách đơn giản hóa như trạng thái biểu diễn được sử dụng trong điều khiển mã hóa, thuật toán tổng hợp khung hình hỗ trợ hai cấu hình. Trong cấu hình đầu tiên, liên quan đến biểu diễn nội suy, một khung hình trung gian được tổng hợp sử dụng cả hai khung hình mã hóa xung quanh. Trong cấu hình thứ hai, liên quan đến biểu diễn không nội suy, một khung hình trung gian được biểu diễn chính từ một khung hình mã hóa, khung hình mã hóa còn lại chỉ được sử dụng cho các vùng biểu diễn mà không hiển thị thấy trong khung hình được mã hóa.

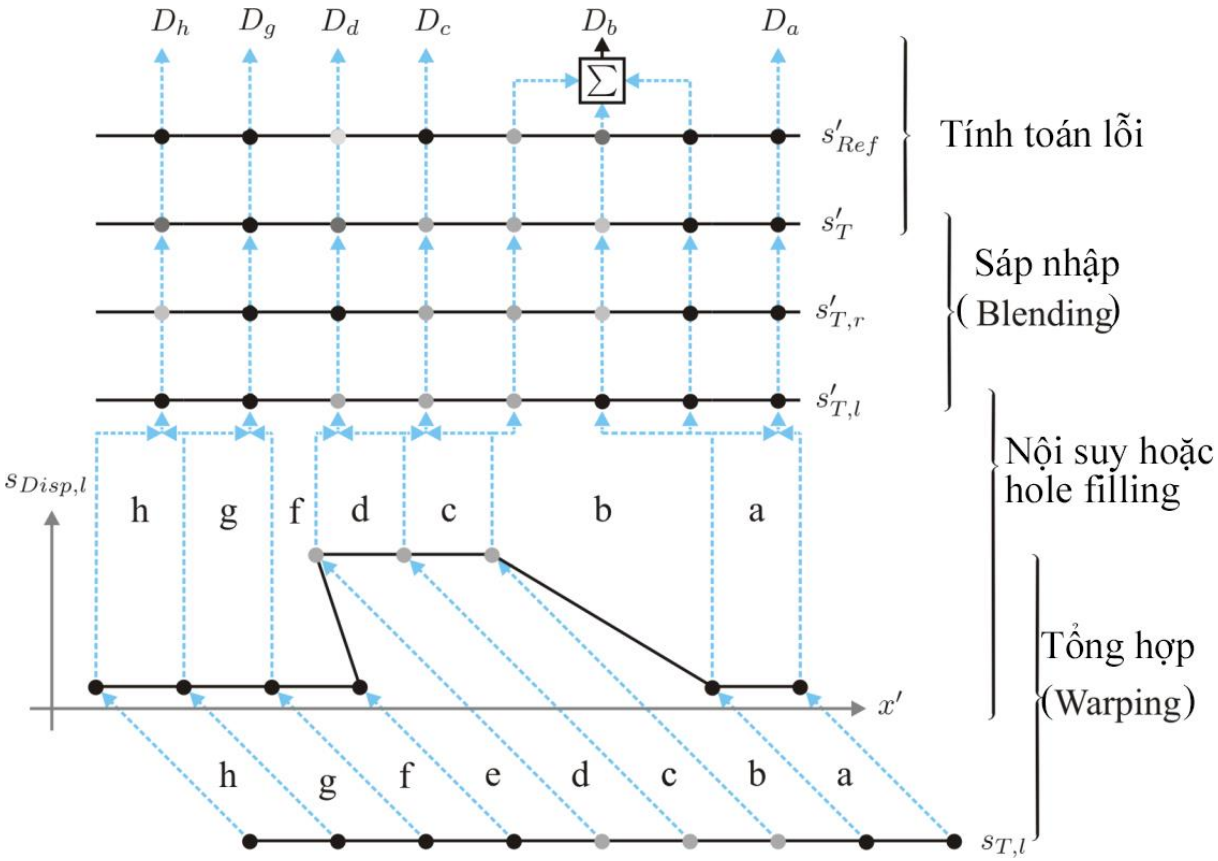
2.5.1. CHUẨN HÓA MẪU

Các vân đầu vào $S_{T,l}$ và $S_{T,r}$ trước tiên được chuẩn hóa để thành $\hat{S}_{T,l}$ and $\hat{S}_{T,r}$ thành phần luma được chuẩn hóa với phân số ¼ theo chiều ngang và thành phần chroma được chuẩn hóa với phân số 1/8 theo chiều ngang và ½ theo chiều dọc. Mục đích của bước này là nhằm giảm đi các hố sinh ra bởi các điểm ảnh được ánh xạ xung quanh. Định dạng đầu vào của VSRS 1D Fast là video YUV 4:2:0 sau khi đầu vào được chuyển đổi thành YUV 4:4:4. Bước này thực hiện đơn giản hóa hơn bởi vì kênh luma và chroma có cùng độ phân giải.

2.5.2. TỔNG HỢP VÀ HOLE FILLING

Tổng hợp, nội suy hay hole filling được thực hiện trong một bước kết hợp. Do các độ lệch tổng hợp được tính toán như được miêu tả trong đầu bước 2. Các bước tổng hợp, nội suy hay hole filling được thực hiện line wise và trong 1 line interval wise. Được xử lý trực tiếp từ trái qua phải. Một khoảng cách trong khung hình đầu ra được xác định bởi các vị trí được tổng hợp x'_s và x'_e của hai mẫu khung hình đầu vào lân cận ở các vị trí x_s và x_e. Sau đó tính toán các khoảng biên, xử lý tiếp tục phụ thuộc vào độ rộng khoảng cách

- **Nội suy** được áp dụng nếu chiều rộng khoảng cách $x'_e - x'_s$ là nhỏ hơn hoặc bằng hai lần khoảng cách mẫu. Quá trình nội suy ở các vị trí giống nhau x'_{FP} được xác định giữa khoảng cách biên x'_s và x'_e được thực hiện. Vì vậy, các mẫu từ phiên bản chuẩn hóa của vân đầu vào $\hat{s}_{T,l}$ được ánh xạ đến vị trí nội suy x'_{FP} trong khung hình tổng hợp $s'_{T,l}$. Vị trí $\hat{x}$ trong khung hình được chuẩn hóa được bắt nguồn từ khoảng cách vị trí nội suy đến khoảng cách biên.



Hình 2.17 : Sự phụ thuộc giữa các tín hiệu đầu vào, trung gian và đầu ra của bước tính toán lỗi, biểu diễn

$$\hat{x} = 4 \cdot \left(\frac{x'_{FP} - x'_s}{x'_e - x'_s} + x_s \right) \qquad (6)$$

- **Disocclusions:** Nếu độ rộng khoảng cách $x'_e - x'_s$ lớn hơn hai lần độ rộng khoảng cách lấy mẫu, một dissocclusion được sinh ra trong khung hình tổng hợp. Thay vào đó hole filling nội suy được thực hiện. Các mẫu này khoảng cách được thiết lập giá trị mẫu phụ thuộc vào khoảng cách biên bên phải $s_{T,l}(x_e)$ (phụ thuộc vào thông tin nền). Các vị trí disocclusion và vị trí lấy mẫu được lấp đầy sẽ được lưu trong bản đồ filling $s'_{F,l}$.

- **Occlusions:** Nếu biên khoảng cách được đảo ngược lại ($x'_e < x'_s$) khoảng cách sẽ được occlude trong khung hình tổng hợp. Biểu diễn ở một vị trí mẫu gần x'_e được thực hiện, nếu khoảng cách kế tiếp là không được occlude và x'_e phụ thuộc vào đối tượng nền. Hơn nữa, thuật toán này sử dụng các đặc tính mà khoảng cách nền occluded sẽ tự động ghi đè lên bởi các đối tượng nền trong khung hình tổng hợp $s'_{T,l}$ do hướng xử lý từ trái qua phải.

Các kênh Chroma của khung hình tổng hợp được biểu diễn cùng với kênh luma và được lưu trong cùng độ phân giải ở đây là luma. Hơn nữa, nếu biểu diễn nội suy được sử dụng, một bản đồ chiều sâu $s'_{D,l}$ được ngoại suy với độ chính xác mẫu đầy đủ từ bản đồ chiều sâu $s_{D,l}$ với các bước được miêu tả như ở trên

2.5.3. TẠO BẢN ĐỒ XÁC THỰC

Trong bước này, bản đồ filling $s'_{F,l}$ được chuyển đổi thành bản đồ xác thực $s'_{R,l}$. Nếu biểu diễn nội suy được sử dụng, các vị trí đánh dấu là discclusion trong $s'_{F,l}$ được ánh xạ tới một độ tin cậy là 0. Trong các phân vùng được xác định bên phải một disocclusion với độ rộng bằng 6 lần mẫu độ tin cậy tăng tuyến tính từ 0 đến 255 từ trái sang phải theo chiều ngang. Tất cả các mẫu khác được gán với một độ tin cậy là 255. Nếu biểu diễn không phải nội suy được sử dụng, các vị trí được đánh dấu như là disocclusion trong $s'_{F,l}$ được ánh xạ tới một độ tin cậy là 0. Tất cả các mẫu khác được gán với độ tin cậy là 255.

2.5.4. TĂNG CƯỜNG TÍNH ĐỒNG NHẤT

Trong bước này biểu đồ histogram của $s'_{T,l}$ tương ứng với histogram của $s'_{T,r}$. Vì vậy, một bảng tìm kiếm (LUT) sẽ thực thi một hàm f được tạo ra, sau đó được áp dụng để lập bản đồ các mẫu của $s'_{T,l}$ tương ứng với giá trị của nó

Hàm f và bảng LUT tương ứng thu được bằng cách giải phương trình tương đương sau

$$h[f(s'_{T,l})] = h[s'_{T,r}] \qquad (7)$$

Ở đây $h[...]$ biểu thị biểu đồ histogram chỉ liên quan đến các mẫu ở các vị trí (x, y) với độ tin cậy $s'_{R,l}(x, y)$ và $s'_{R,r}(x, y)$. Các kênh chroma được xử lý tương tự theo cách này

2.5.5. KẾT HỢP

$s'_{T,l}$ và $s'_{T,r}$ được kết hợp để tạo ra khung hình tổng hợp đầu ra trong bước này. Trong trạng thái biểu diễn nội suy được sử dụng, việc quyết định cách pha trộn được thực hiện phụ thuộc vào bản đồ độ tin cậy $s'_{R,l}$ hoặc $s'_{R,r}$ và bản đồ độ sâu được biểu diễn $s'_{D,l}$ và $s'_{D,r}$. Các quy tắc để xác định giá trị mẫu sáp nhập $s'_T(x,y)$ từ $s'_{T,l}(x,y)$ và $s'_{T,r}(x,y)$ được thực hiện như dưới đây:

- Nếu vị trí (x,y) được disocclude (độ tin cậy 0) chỉ trong một khung hình, giá trị mẫu từ khung hình khác được sử dụng.
- Mặt khác, nếu giá trị (x,y) được disocclude trong cả hai khung hình, giá trị mẫu cuối cùng được sử dụng.
- Mặt khác, nếu sự khác nhau về độ sâu hình thành từ $s'_{D,l}(x,y)$ và $s'_{D,r}(x,y)$ ở trên là một giá trị ngưỡng, mẫu trước đó được sử dụng.
- Mặt khác, nếu một mẫu là không đáng tin cậy với một giá trị 255, một trọng số trung bình với độ tin cậu đưa ra là trọng số được sử dụng
- Mặt khác, một trọng số trung bình giữa $s'_{T,l}(x,y)$ và $s'_{T,r}(x,y)$ với một trọng số cao hơn cho khung hình gần hơn vị trí khung hình ảo được sử dụng

Nếu trạng trái biểu diễn không phải là nội suy được sử dụng, khung hình trung gian được biểu diễn chính từ một khung hình được sử dụng và chỉ có các hố được lấp đầy từ khung hình khác. Giả sử rằng $s'_{T,l}$ là khung hình chính, các quy tắc cho việc xác định giá trị mẫu $s'_T(x,y)$ từ $s'_{T,l}(x,y)$ và $s'_{T,r}(x,y)$ được thực hiện như sau:

- Nếu $s'_{R,l}(x,y)$ bằng 255 hoặc $s'_{R,r}(x,y)$ bằng 0, giá trị mẫu $s'_{T,l}(x,y)$ được sử dụng
- Mặt khác, nếu $s'_{R,l}(x,y)$ bằng 0, giá trị mẫu $s'_{T,r}(x,y)$ được sử dụng
- Mặt khác, một trọng số trung bình với độ tin cậy được đưa ra như là trọng số được sử dụng

CHƯƠNG 3: THUẬT TOÁN HOLE FILLING SWA

3.1. GIỚI THIỆU THUẬT TOÁN

Nhằm tăng cường chất lượng cho khung hình ảo 3D, xóa bỏ các nhiễu biên, hình giả và lấp đầy các vùng hố sinh ra trong quá trình tổng hợp khung hình ảo. Thực tế, có rất nhiều thuật toán Hole filling được đề xuất, Tuy nhiên kết quả thực nghiệm cho thấy rằng các thuật toán đều không cho hiệu quả rõ rệt. Một số thuật toán thì chỉ lấp đầy các hố có kích thước nhỏ, một số chỉ loại bỏ được nhiễu sinh ra trong quá trình tổng hợp.
Trong số các thuật toán cho kết quả tốt nhất hiện nay, thuật toán Hole filling SWA (Spiral weighted average algorithm) [6] có nhiều ưu điểm hơn cả. Thuật toán Hole filling SWA trước tiên loại bỏ nhiễu biên, tìm các vùng occlusion và mở rộng các vùng này đến vùng lỗ trống trong khung hình tổng hợp. Sau đó, thuật toán xác định các giá trị điểm ảnh của các hố dựa trên thuật toán trọng số trung bình đường xoắn ốc và thuật toán tìm kiếm gradient. Thuật toán trọng số trung bình đường xoắn ốc giữ lại biên của từng đối tượng tương đối tốt với thông tin về chiều sâu. Tuy nhiên, nhược điểm của thuật toán này mang lại một hiệu ứng màu xung quanh các hố, dẫn đến chất lượng video không tốt. Để giải quyết vấn đề này, thuật toán tìm kiếm gradient sẽ giữ lại các thành phần tần số cao để giữ các chi tiết trong khung hình tổng hợp. Mặt hạn chế khác của thuật toán này là sinh ra các điểm khiếm khuyết. Để loại bỏ các điểm khiếm khuyết, thuật toán Hole filling SWA sẽ sử dụng một mặt nạ xác suất

3.2. CÁC BƯỚC THỰC HIỆN TRONG THUẬT TOÁN HOLE FILLING SWA

3.2.1. PHÁT HIỆN NHIỄU BIÊN

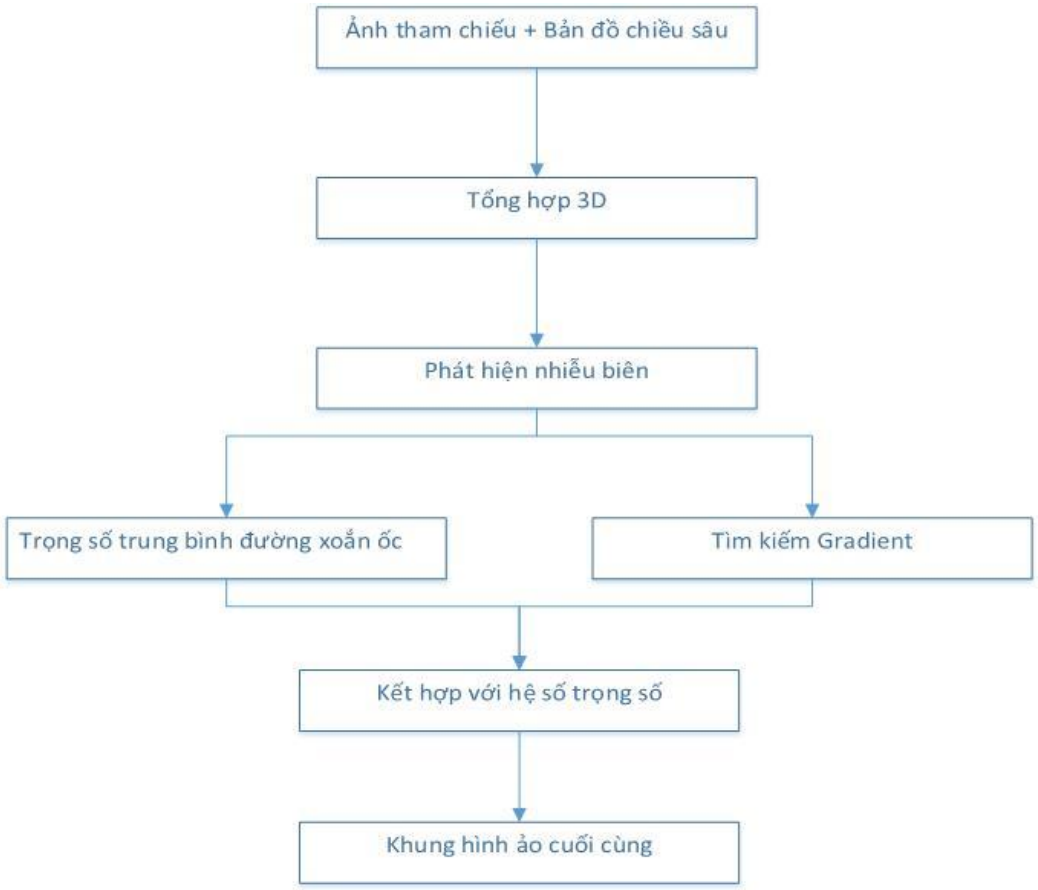
Như kết quả của tổng hợp 3D trong quá trình tổng hợp khung hình, nhiễu biên vẫn còn bởi vì độ không chính xác giữa biên của bản đồ chiều sâu và ảnh vân trong khung hình tham chiếu đưa ra. Ví dụ nhiễu biên được chỉ ra trong Hình 3.1. Ta dễ dàng thấy được các thành phần biên còn lại bên trong vòng tròn trong Hình 3.1. Hình dáng nhiễu biên dường như trở thành một vật được chỉ ra trong hình và nó xảy ra xung quanh đối tượng

Hình 3.1: Nhiễu biên [6]

Hình 3.2: Các hố chung [6]

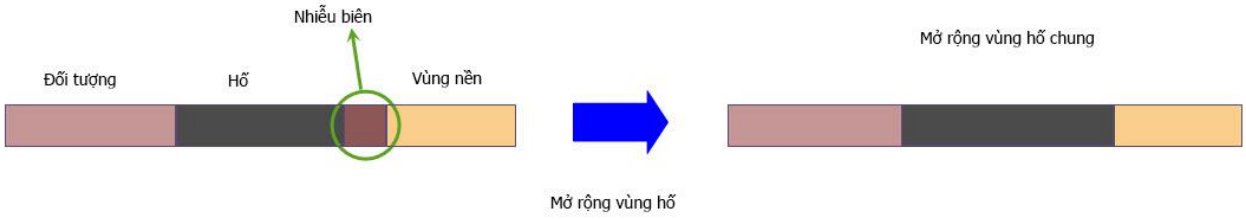
Các hố chung được tạo ra trong khi tổng hợp kết quả khung hình ảo, Nhìn chung, các hố chung này bao phủ các vùng lân cận trong các khung hình tham chiếu tổng hợp. Tuy nhiên, các hố chung này khó khăn để phục hồi hoàn toàn. Hình 3.2 chỉ ra một ví dụ về các hố chung

Nhiễu biên xảy ra do sự sai lệch biên giữa độ sâu và vân ảnh trong 3D tổng hợp



Hình 3.3 : Sơ đồ khối thuật toán Hole filling SWA

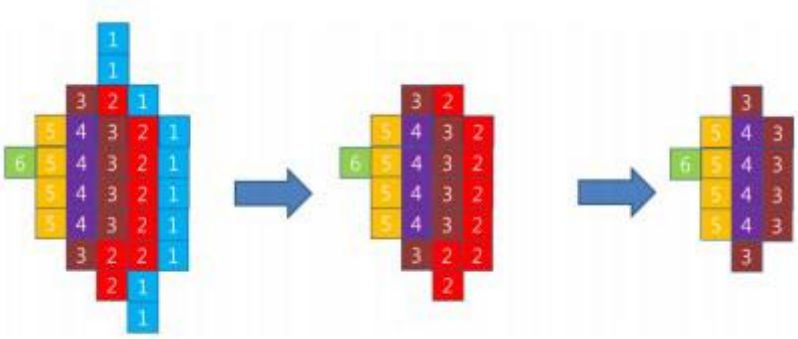
Nếu hố chung được lấp đầy mà không loại bỏ nhiễu thì chất lượng của các hố lấp đầy bị xấu đi bởi vì thông tin màu sắc của nhiễu biên được sử dụng để lấp đầy các hố chung. Trong thuật toán Hole filling SWA, chúng ta tìm thấy trung bình của vùng cố định từ vùng nền. Sau đó chúng ta so sánh giá trị trung bình này với các giá trị điểm ảnh vùng gần nhất và xem xét nhiễu biên khi sự khác biệt tuyệt đối là lớn hơn một ngưỡng. Tính đến trường hợp nhiễu biên liên tục xuất hiện, nếu giá trị điểm ảnh đầu tiên được xác định là nhiễu biên thì chúng ta so sánh giá trị điểm ảnh tiếp theo để phát hiện nhiễu biên liên tục. Vùng nhiễu biên tiếp xúc được gán thành vùng hố và được loại bỏ. Khi so sánh với hình ảnh được lấp đầy mà không loại bỏ nhiễu biên thì hình ảnh thông qua thuật toán Hole filling SWA cho kết quả chất lượng hơn



Hình 3.4: Thuật toán Hole filling SWA loại bỏ nhiễu biên

3.2.2. XÁC ĐỊNH THƯ TỰ HOLE FILLING ĐỐI VỚI VÙNG NỀN

Hình 3.5 chỉ tiến trình của thuật toán hole filling, bắt đầu với các điểm ảnh ngoài cùng và kết thúc ở trung tâm. Trong trường hợp này, thông tin màu sắc của đối tượng gần các vùng hố được sử dụng và chất lượng của kết quả hình ảnh sẽ thấp hơn. Vì vậy trong thuật toán Hole filling SWA, sẽ lấp đầy các vùng nền trước. Hình dưới là thứ tự filling trong trường hợp khung hình điểm ảo bên phải khung hình điểm tham chiếu . Trong trường hợp này, khung hình điểm ảo nằm bên phải khung hình điểm tham chiếu, các hố chung xuất hiện bên phải đối tượng. Do đó, vùng bên phải hố chung trở thành vùng nền và hố chung được lấp đầy từ điểm đầu tiên này (hình 3.6a). Trong hình 3.6b, ta có thể thấy rằng thuật toán Hole filling SWA là ít ảnh hưởng bởi thông tin đối tượng hơn là các thuật toán tồn tại



(a)

(b)

Hình 3.5 : (a) Thứ tự thuật toán Hole filling SWA; (b) Kết quả

3.2.3. THUẬT TOÁN TRỌNG SỐ TRUNG BÌNH ĐƯỜNG XOẮN ỐC

Tiến trình thuật toán trọng số trung bình đường xoắn ốc trong Hình 3.6 được chỉ ra như dưới đây:

(1) Trước tiên, tìm các đường biên bên trong của vùng hố. Sau đó, chọn một điểm ảnh từ ranh giới biên này và bắt đầu tiến trình filling ở điểm ảnh này.

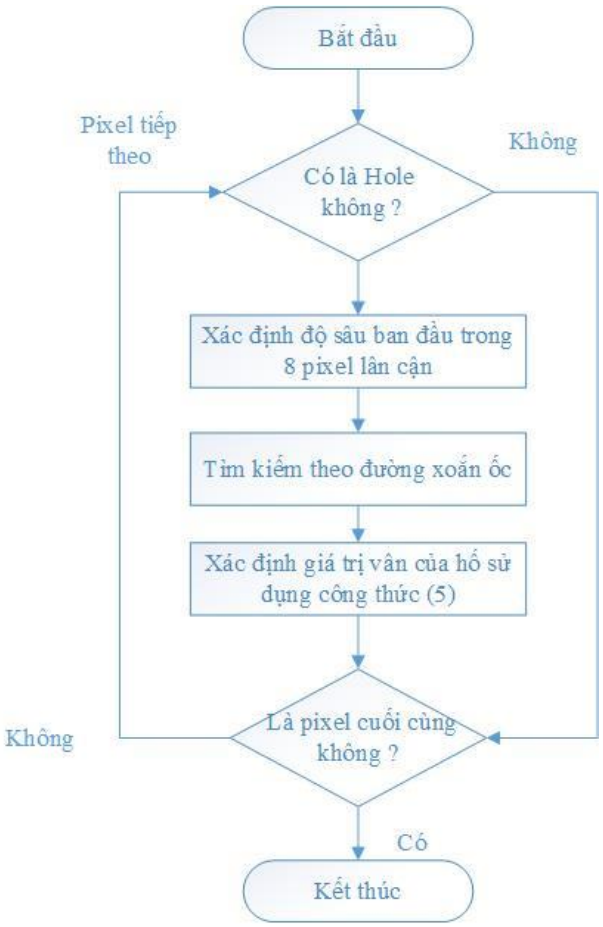
(2) Các điểm ảnh hố ban đầu được chọn trong bước (1) được gán với giá trị độ sâu nhỏ nhất của 8 giá trị lân cận mà không phụ thuộc vào vùng hố chung. Thực hiện tìm kiếm theo đường xoắn ốc ở vị trí hố chung ban đầu theo thứ tự tìm kiếm

(3) Trong quá trình tìm kiếm theo đường xoắn ốc, trọng số vân và các giá trị độ sâu của các điểm ảnh với trọng số khác nhau phụ thuộc vào khoảng cách giữa điểm ảnh ban đầu với điểm ảnh hiện tại được lưu nếu giá trị khác nhau về độ sâu phụ thuộc vào khoảng cách giữa vân ban đầu và vân hiện tại là nhỏ hơn một ngưỡng. Giá trị trung bình của tất cả các giá trị trọng số vân và giá trị độ sâu của điểm ảnh ban đầu sau đó sẽ được gán lại như là giá trị vân và độ sâu của điểm ảnh ban đầu. Tiến trình này được gọi là trọng số trung bình theo đường xoắn ốc và được biểu diễn trong công thức

$$ST_{(x,y,t)} = \sum_{(p,q)\in SR} \left(\frac{e(p,q)T(p,q,t)}{E} \right) \qquad (\mathbf{8})$$

$$SD_{(x,y,t)} = \sum_{(p,q)\in SR} \left(\frac{e(p,q)d(p,q,t)}{E} \right) \qquad (\mathbf{9})$$

$$E = \sum_{(p,q)\in SR} e(p,q), e(p,q) = \frac{D(p,q)}{W(p,q)} \qquad (\mathbf{10})$$



Hình 3.6: Biểu đồ luồng thuật toán trọng số trung bình đường xoắn ốc

$$D(p,q) = \begin{cases} 1, & \left|Initial_{depth} - d(p,q) < th\right| \\ 0, & else \end{cases} \quad (\mathbf{11})$$

Trong đó :

$T(p,q,t)$ và $d(p,q,t)$ là giá trị vân và giá trị độ sâu được lưu của điểm ảnh (p,q) trong thứ tự tìm kiếm SR

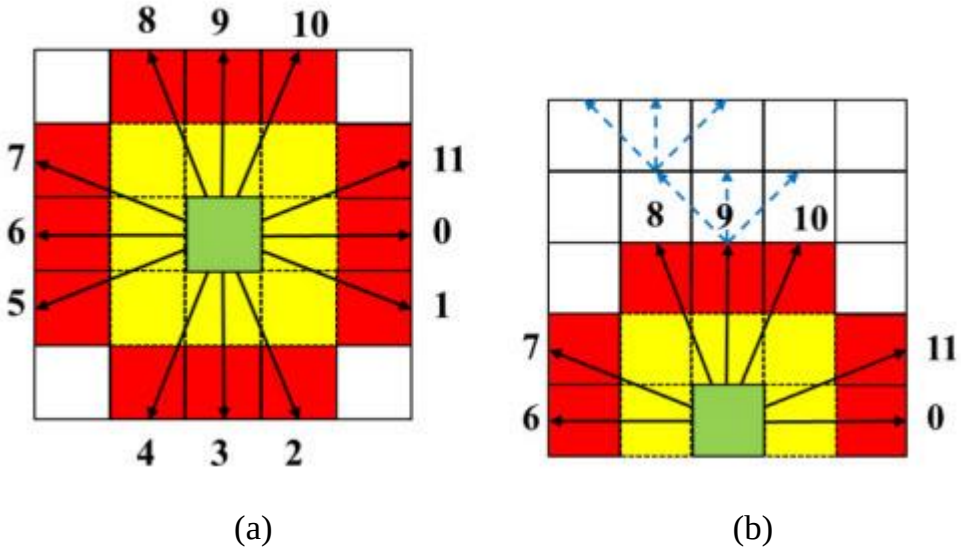
$ST_{(x,y,t)}$ *và* $SD_{(x,y,t)}$ là giá trị vân và giá trị độ sâu được gán mới của hố tại vị trí (x,y) tương ứng

$D(p,q)$ là sự khác nhau về giá trị độ sâu giữa điểm ảnh ban đầu tại vị trí (x,y) và điểm ảnh hiện tại tại vị trí (p,q)

W(*p*,*q*) chỉ ra hệ số trọng số, là khoảng cách Euclidian giữa 2 điểm ảnh (*x*,*y*) và (*p*,*q*).

3.2.4. THUẬT TOÁN TÌM KIẾM GRADIENT

Thuật toán trọng số trung bình đường xoắn ốc sẽ tạo ra hiệu ứng màu lan truyền do đặc tính bộ lọc băng thông thấp. Để giải quyết vấn đề này, chúng ta sử dụng thông tin gradient. Thông tin gradient có thể lưu trữ các chi tiết trong khung hình tổng hợp



(a) (b)

Hình 3.7: Thuật toán tìm kiếm Gradient, bước (1) và bước (2)

(1) Trước tiên, tính toán sự khác biệt cường độ giữa một điểm ảnh và 8 điểm ảnh lân cận (vùng đánh dấu màu vàng trong hình 3.7a của hố ban đầu và một điểm ảnh từ các điểm ảnh liền kề trong 12 hướng (vùng màu đỏ trong hình 3.7a). Sau đó xác định điểm ảnh với giá trị khác nhau lớn nhất từ các điểm ảnh lân cận (vùng đỏ trong hình 3.7a)

(2) Tiếp theo, lặp lại bước (1) ở các điểm ảnh được lựa trọn trong bước (1), nhưng chỉ theo 3 hướng đơn giản, được minh họa trong Hình 3.7b

(3) Lặp lại bước (2) cho tất cả các điểm ảnh trong phạm vi tìm kiếm được xác định trước

(4) Cuối cùng, giá trị của một hố được gán giá trị trung bình của các điểm ảnh được xác định ở bước (1) và bước (3)

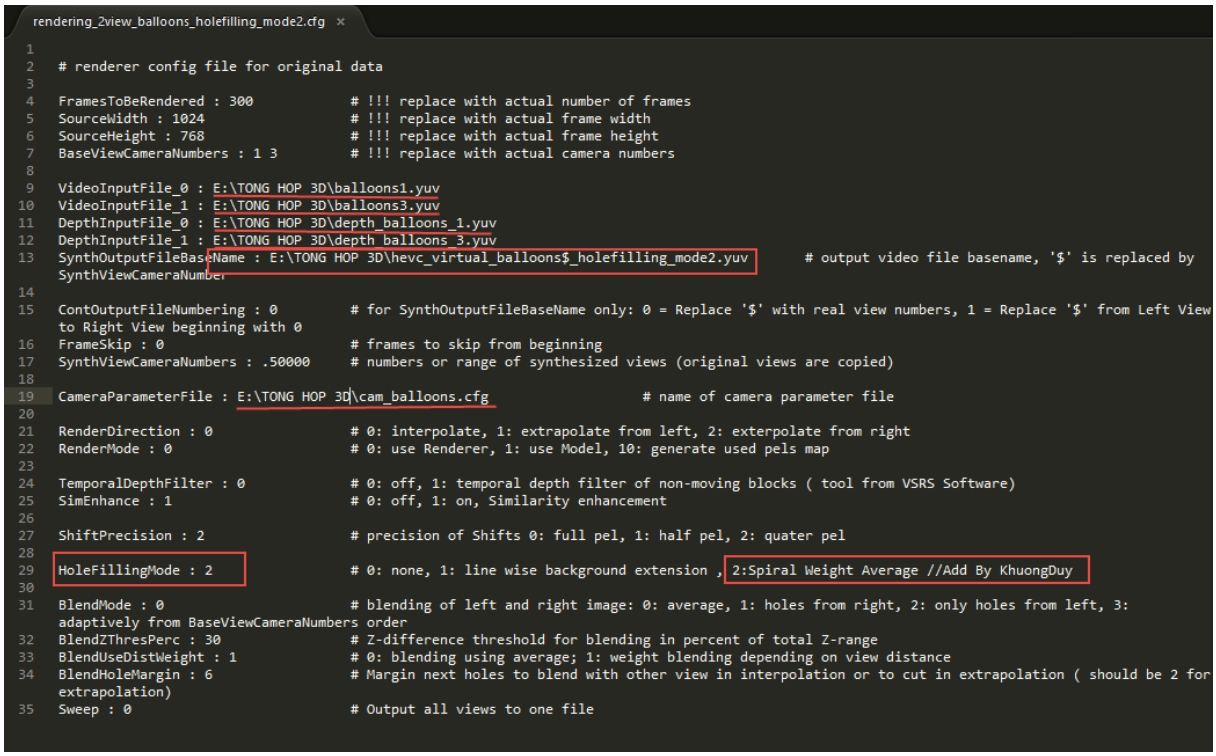
CHƯƠNG 4: CÀI ĐẶT VÀ KẾT QUẢ THỰC NGHIỆM

4.1. CÀI ĐẶT THỰC NGHIỆM

Luận văn đã tiến hành thực nghiệm dựa trên thuật toán Hole filling SWA (được trình bày trong Chương 3) trên 7 chuỗi đa khung hình được xác định trong thực nghiệm với phần mềm tham chiếu 3D-HEVC**:** Pantomime**,** Balloons, Kendo, Lovebird, Newspaper, Cafe và Champagne. Với mỗi chuỗi, chúng tôi xem xét 2 khung hình tham chiếu và tổng hợp thành một khung hình giữa với thuật toán Hole filling SWA với phần mềm tham chiếu VSRS1D-Fast của 3D-HEVC, thuật toán VSRS trong phần mềm VSRS 4.0. Để có được kết quả so sánh khách quan, luận văn sử dụng thuật toán Hole filling với tập dữ liệu mẫu, ảnh độ sâu như nhau. Bảng 4.1 chỉ ra chi tiết các chuỗi trong thực nghiệm. Luận văn đánh giá PSNR các khung hình tổng hợp so với các khung hình ban đầu của mỗi chuỗi kiểm thử. Quá trình được thực hiện với 3D-HEVC, khung hình trái được thiết đặt như là khung hình cơ bản và khung hình phải được coi như khung hình độc lập.

Độ phân giải Rộng*Cao	**Chuỗi**	**Số lượng khung hình**	**Các khung hình**
1280x 960	Pantomime	250	37 39 41
	Champagne	150	
1024x 768	Balloons	150	1 3 5
	Kendo	150	1 3 5
	Lovebird	300	4 6 8
	Newspaper	300	2 4 6
1920x1080	Cafe	300	2 4 6

Bảng 4.1 : Các chuỗi được sử dụng trong thí nghiệm



Hình 4.1: File cấu hình chương trình .cfg

Chương trình đầu vào sau khi biên dịch ra TAppRenderer.exe. Để chạy chương trình tạo ra khung hình ảo. Ta thực hiện như cú pháp dưới đây:

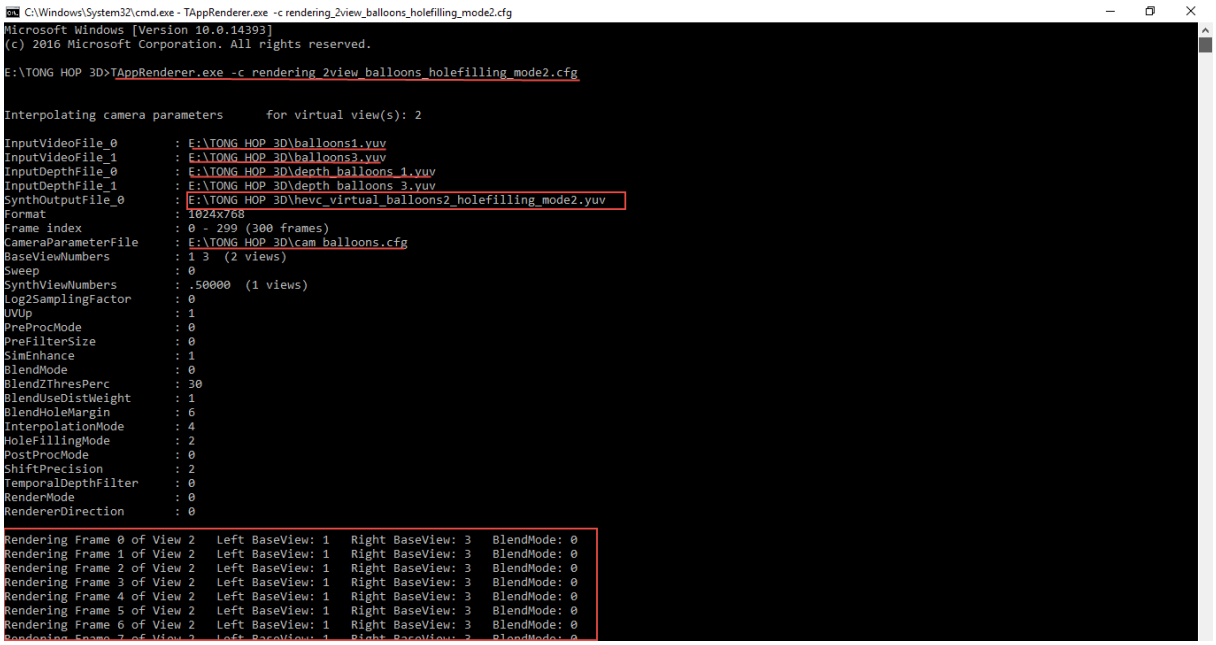
TAppRenderer.exe [-c config.cfg]

Trong đó:

TAppRenderer.exe: Tên chương trình được biên dịch

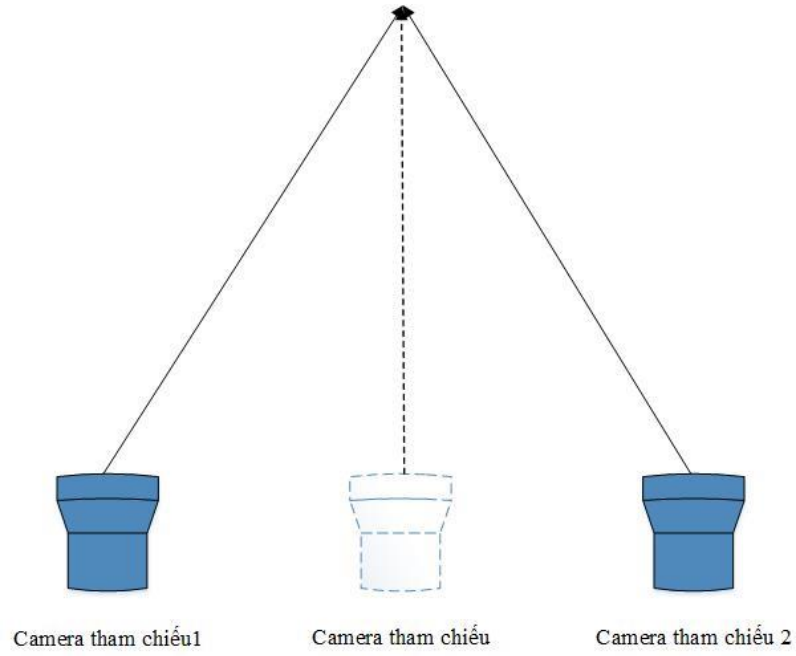
-c: Định nghĩa file cấu hình được sử dụng. Nhiều file cấu hình, tham số -c được lặp lại

Config.cfg : File cấu hình như Hình 4.1



Hình 4.2 : Giao diện chạy chương trình

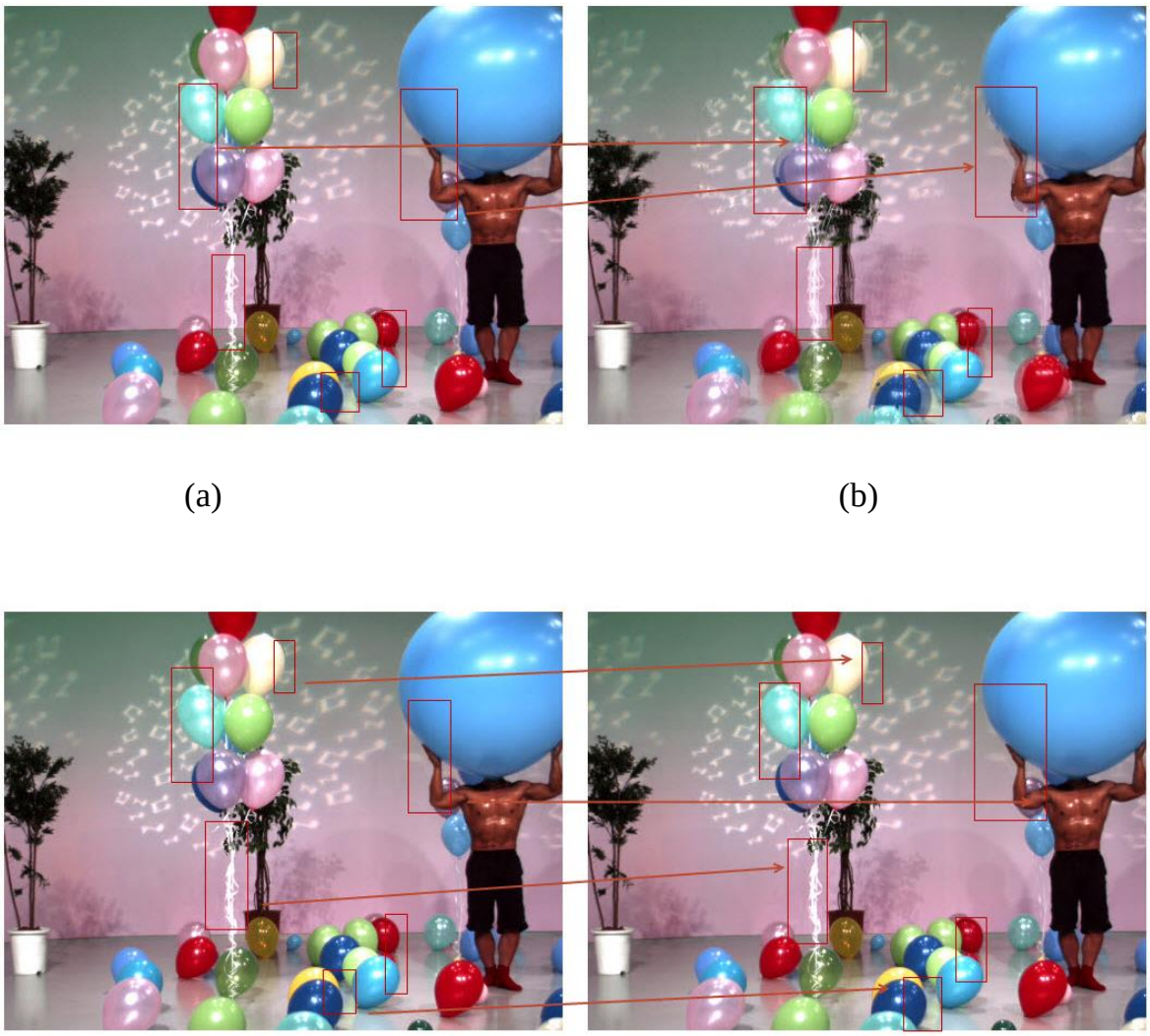
4.2. KẾT QUẢ TỔNG HỢP KHUNG HÌNH



Hình 4.3: Tổng hợp khung hình trong trường hợp nội suy

Trong Hình 4.1 chỉ ra trường hợp tổng hợp khung hình nội suy. Kết quả sinh ra một hình ảnh điểm quan sát ảo ở vị trí phía trong của các khung hình tham chiếu bằng thuật toán Hole filling SWA và so sánh hiệu năng của nó so với thuật toán Hole filling trong VSRS 4.0 anpha; thuật toán Hole filling trong 3D-HEVC.

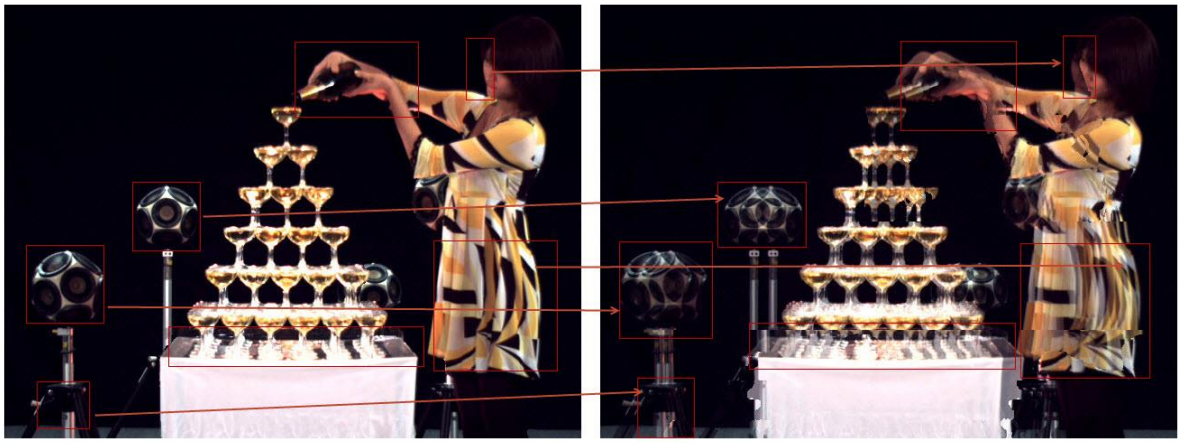
Các kết quả hình ảnh dưới đây được chụp từ các khung hình tổng hợp lên khi chạy thực nghiệm trong thí nghiệm:



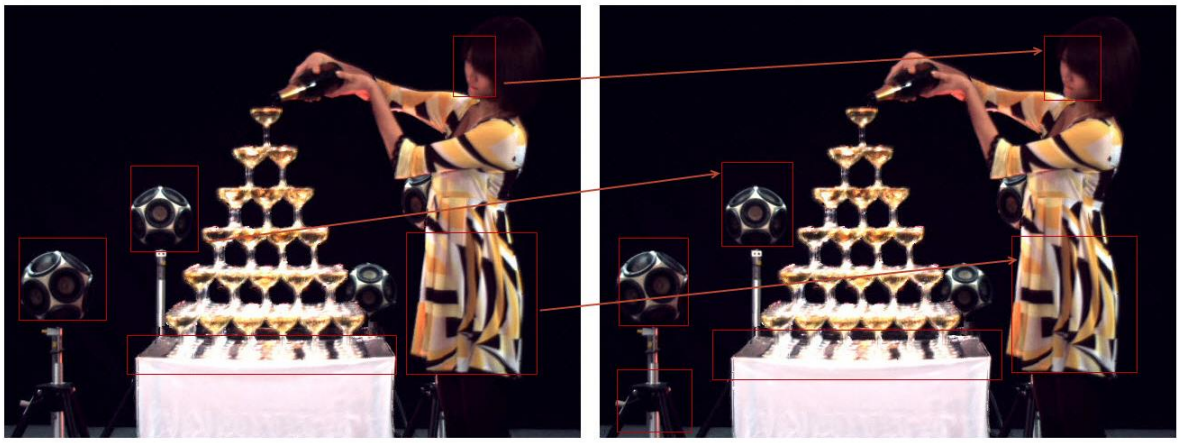
(a) (b)

(c) (d)

Hình 4.4: Khung hình ảo tổng hợp - "Balloon"; (Khung hình thứ 2) (a): VSRS3.5; (b): VSRS4.0; (c): 3D-HEVC ; (d) Thuật toán Hole filling SWA

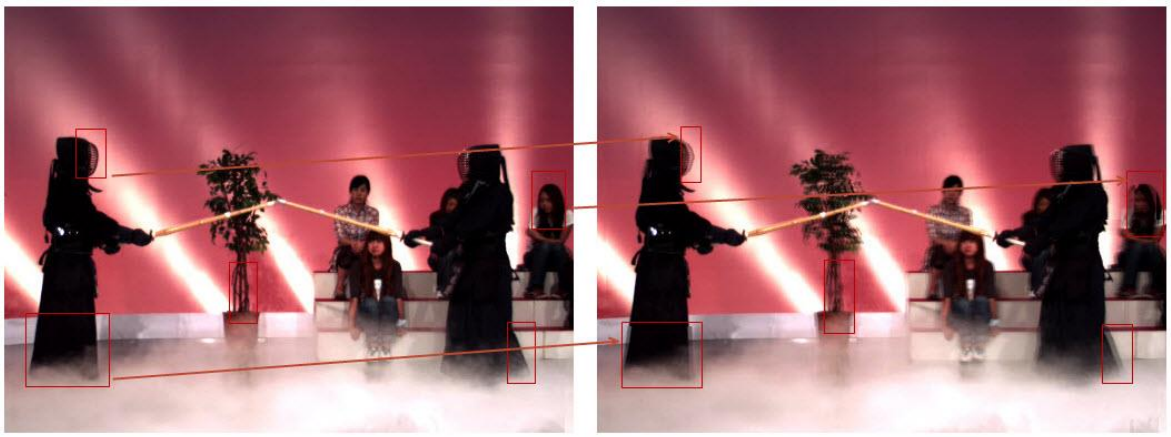


(a) (b)

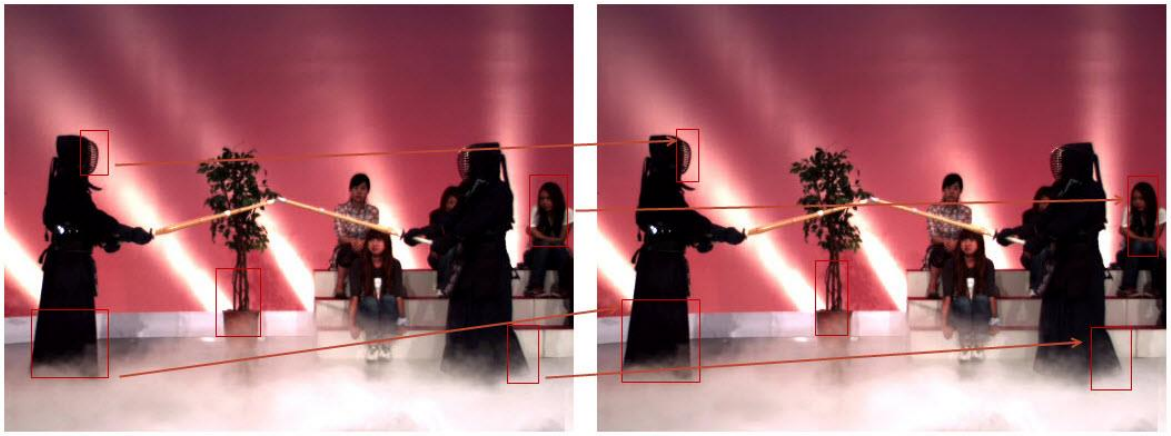


(c) (d)

Hình 4.5: Khung hình ảo tổng hợp - "Champagne" (Khung hình 38) (a): VSRS3.5; (b): VSRS4.0 ;(c): 3D-HEVC ; (d) Thuật toán Hole filling SWA

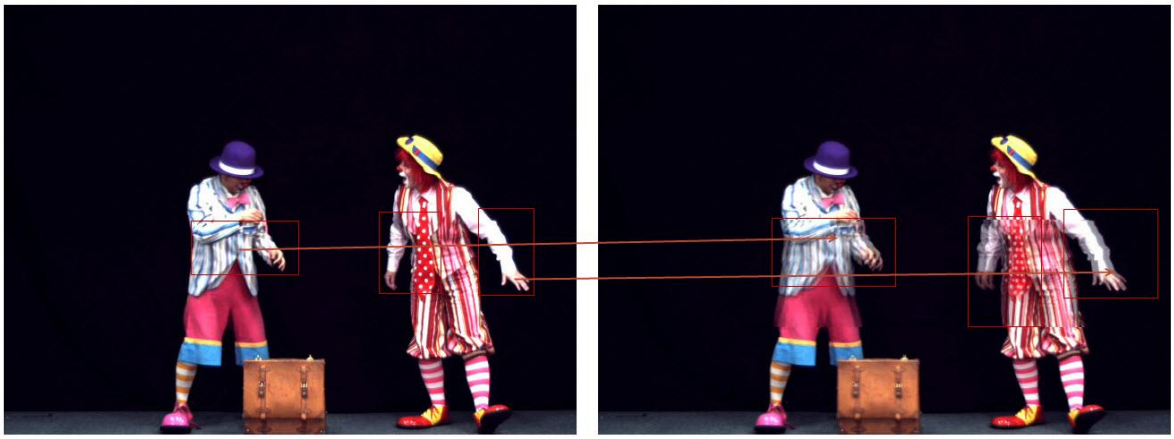


(a) (b)

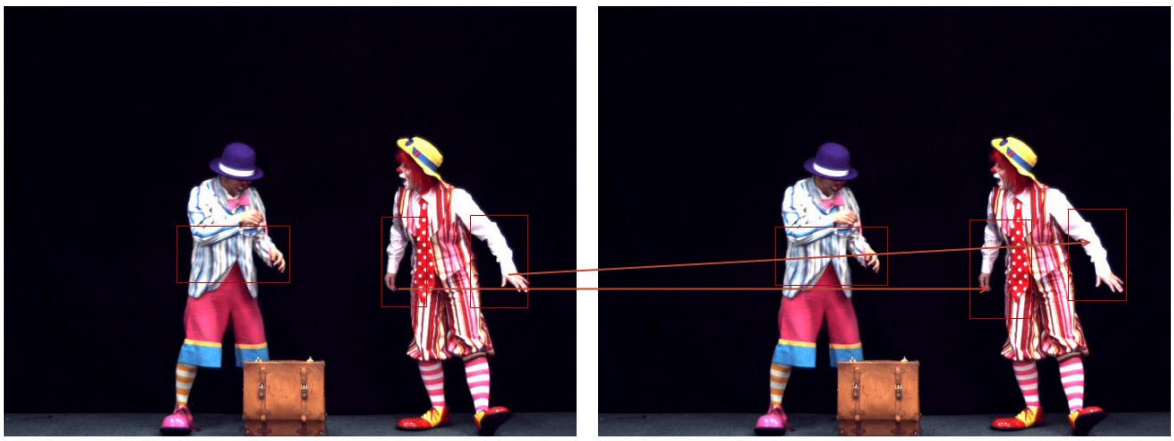


(c) (d)

Hình 4.6: Khung hình ảo tổng hợp - “Kendo”; (Khung hình thứ 2) (a): VSRS3.5; (b): VSRS4.0; (c): 3D-HEVC ; (d) Thuật toán Hole filling SWA



(a) (b)



(c) (d)

Hình 4.7: Khung hình ảo tổng hợp - "Pantomime" (Khung hình 38)

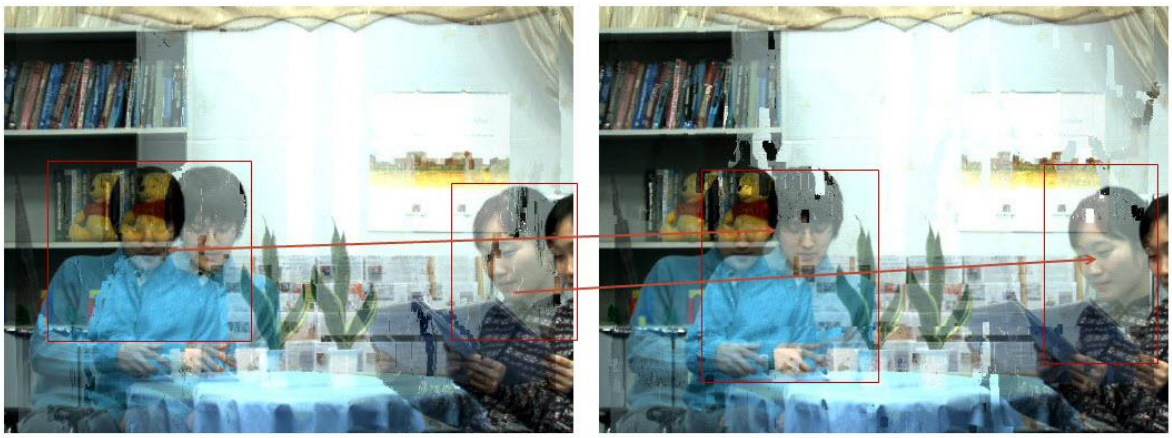
(a): VSRS3.5; (b): VSRS4.0; (c): 3D-HEVC ; (d) Thuật toán Hole filling SWA

(a) (b)

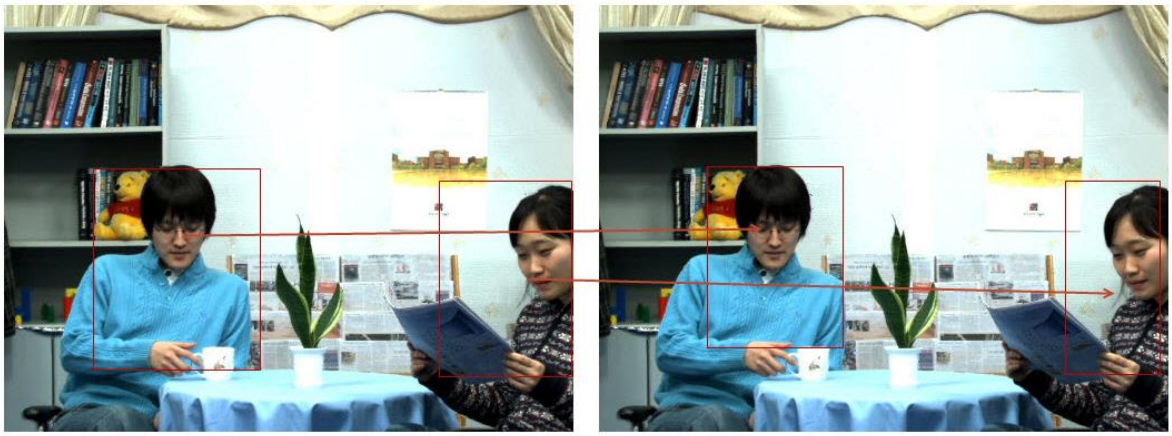
(b) (d)

Hình 4.8: Khung hình ảo tổng hợp - "Lovebird" (Khung hình 7)

(a): VSRS3.5; (b): VSRS4.0; (c): 3D-HEVC ; (d) Thuật toán Hole filling SWA



(a) (b)



(c) (d)

Hình 4.9: Khung hình ảo tổng hợp - "Newspaper" (Khung hình 3)

(a): VSRS3.5; (b): VSRS4.0; (c): 3D-HEVC ; (d) Thuật toán Hole filling SWA

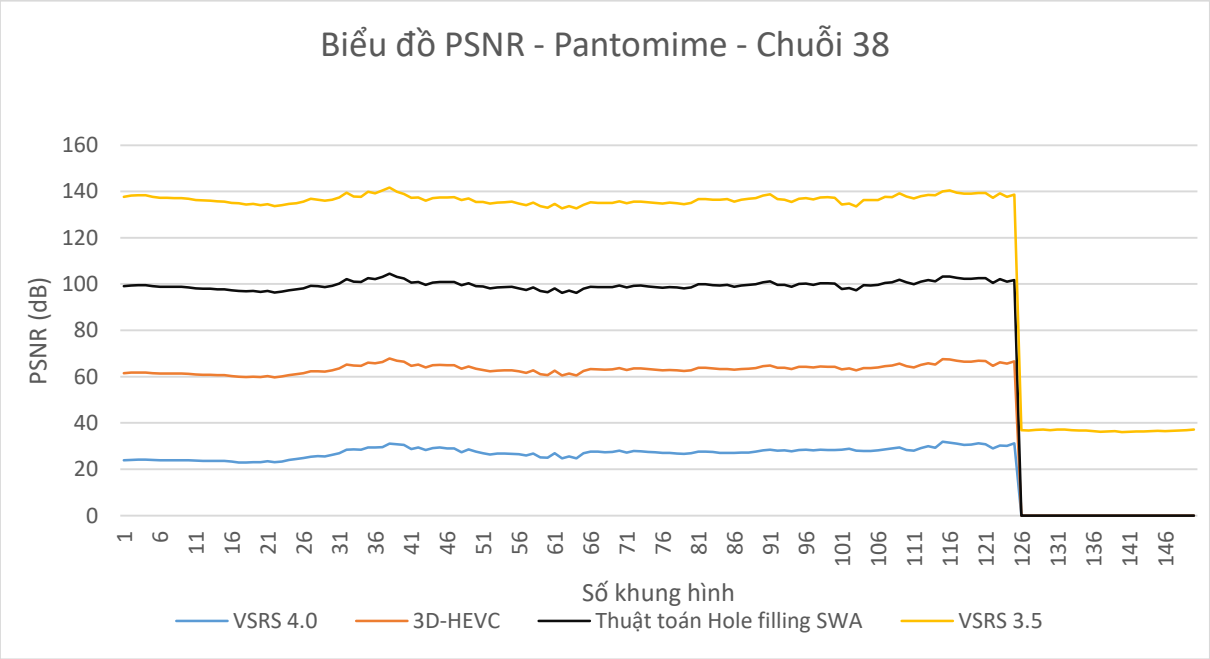
Tất cả dữ liệu của các chuỗi kiểm thử trong bảng là những giá trị trung bình cho tất cả các khung hình của mỗi chuỗi, như được chỉ ra trong Bảng 4.2. Tương tự như vậy, số lượng khung hình tham chiếu chỉ ra các vị trí của các camera tham chiếu. Và số lượng khung hình ảo chỉ ra vị trí của một camera ảo. Trong Hình 4.1 và 4.2, thuật toán Hole filling SWA thực hiện tốt hơn so với các thuật toán khác cũng trong các trường hợp nội suy. Chúng ta có thể thấy rằng thuật toán Hole filling SWA cho kết quả tốt hơn các thuật toán khác trong vùng màu đỏ đánh dấu trong cả hai hình 4.1 và 4.2

Tên chuỗi kiểm thử	**VSRS 3.5**	**VSRS 4.0**	**3D-HEVC**	**Thuật toán Hole filling SWA**
Pantomime	36.92823	26.56432	36.14031	36.15579
Balloons	36.22950	26.50836	36.11454	36.17031
Kendo	35.77918	29.82488	36.36218	36.00484
Champagne	34.09464	19.08684	28.69579	28.72952
Lovebird	20.62912	23.80880	27.92411	27.89786

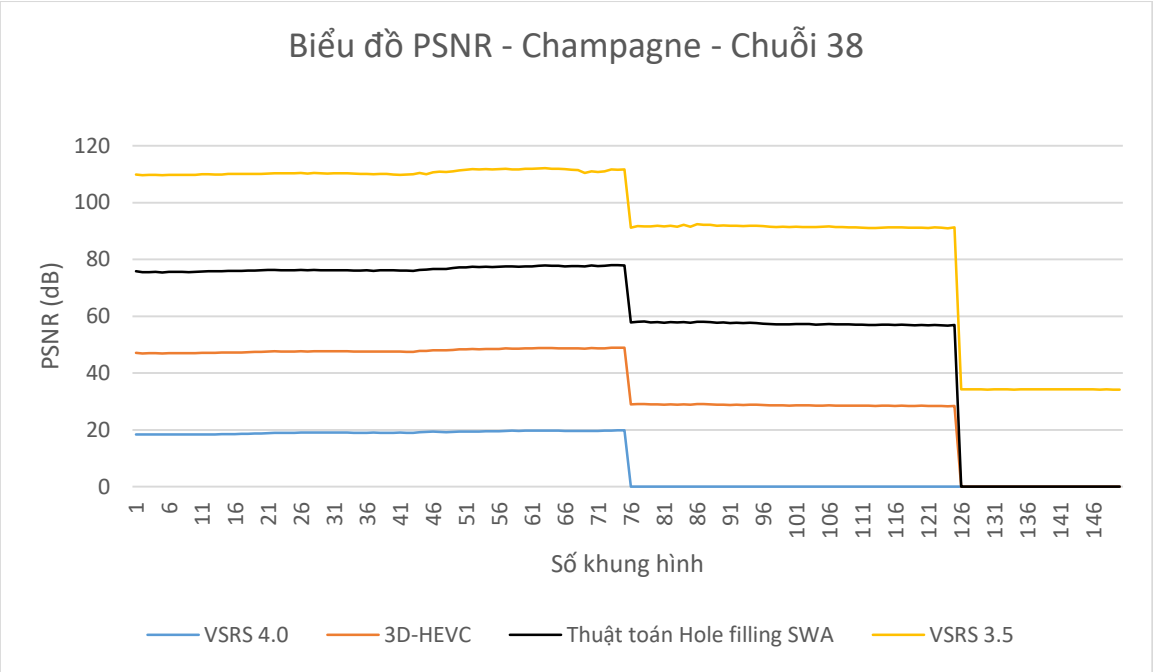
Newspaper	14.68198	15.04415	32.01516	31.81163

Bảng 4.2: So sánh hiệu năng PSNR giữa các thuật toán trong các phần mềm

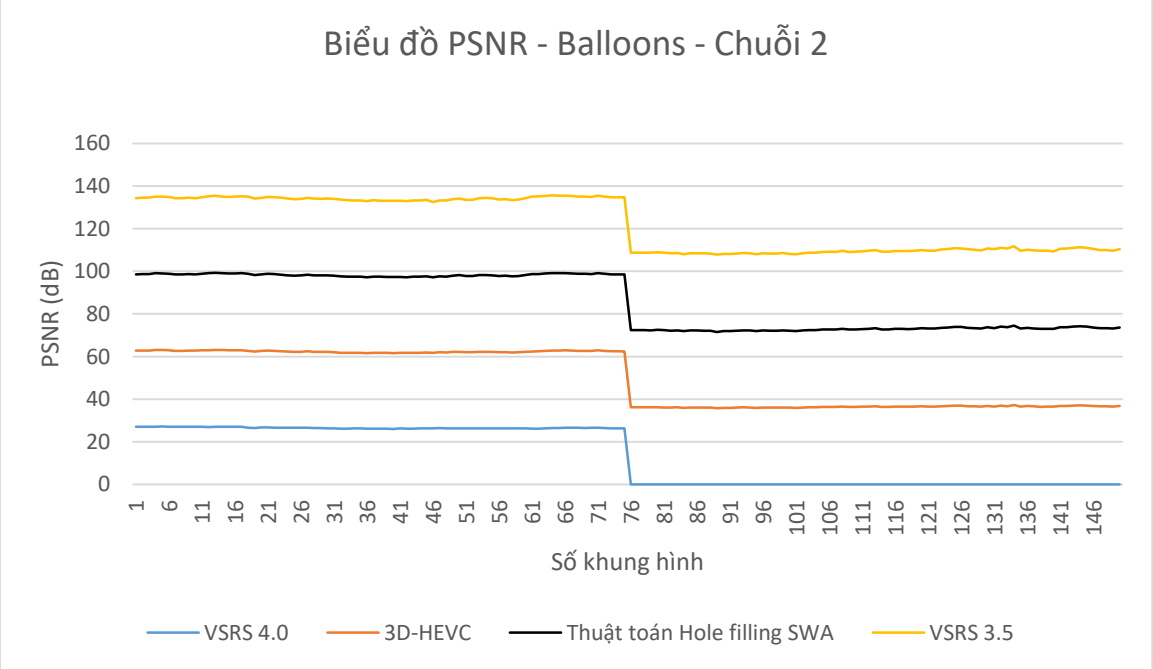
Trong Bảng 4.2, chúng ta so sánh hiệu năng PSNR của thuật toán Hole filling SWA trong trường hợp nội suy. Bảng 4.2 chỉ ra rằng nhìn chung thuật toán Hole filling SWA cho kết quả tốt hơn tuy có 1 vài chuỗi kết quả không tốt bằng nhưng sự khác biệt không phải là lớn



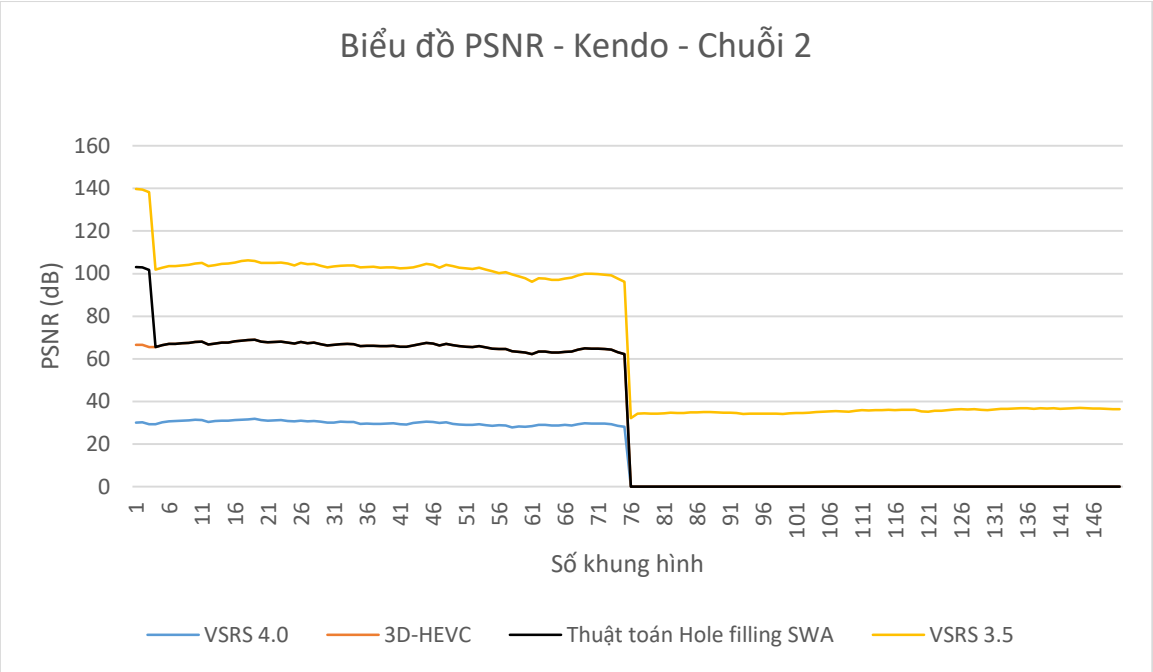
(a)


Biểu đồ PSNR - Champagne - Chuỗi 38
PSNR (dB)
120
100
80
60
40
20
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96 101 106 111 116 121 126 131 136 141 146
Số khung hình
VSRS 4.0
3D-HEVC
Thuật toán Hole filling SWA
VSRS 3.5

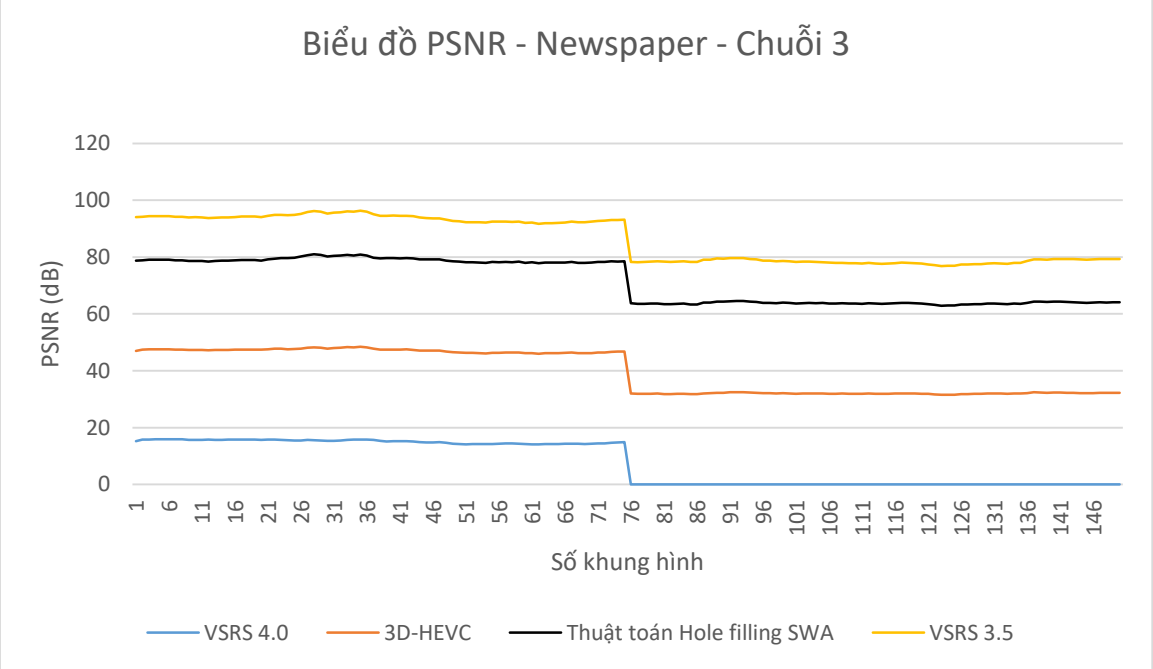
(b)


Biểu đồ PSNR - Balloons - Chuỗi 2
PSNR (dB)
160
140
120
100
80
60
40
20
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96 101 106 111 116 121 126 131 136 141 146
Số khung hình
VSRS 4.0
3D-HEVC
Thuật toán Hole filling SWA
VSRS 3.5

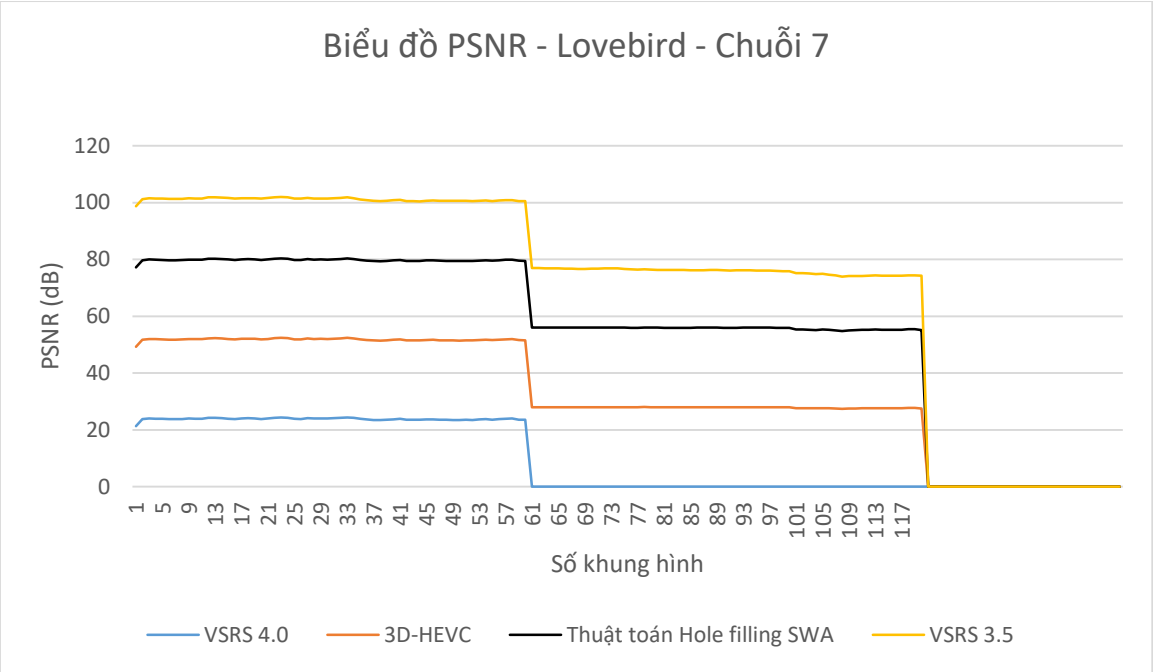
(c)


Biểu đồ PSNR - Kendo - Chuỗi 2
PSNR (dB)
160
140
120
100
80
60
40
20
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96 101 106 111 116 121 126 131 136 141 146
Số khung hình
VSRS 4.0
3D-HEVC
Thuật toán Hole filling SWA
VSRS 3.5

(d)


Biểu đồ PSNR - Newspaper - Chuỗi 3
PSNR (dB)
120
100
80
60
40
20
0
1 6 11 16 21 26 31 36 41 46 51 56 61 66 71 76 81 86 91 96 101 106 111 116 121 126 131 136 141 146
Số khung hình
VSRS 4.0
3D-HEVC
Thuật toán Hole filling SWA
VSRS 3.5

(e)



(f)

Hình 4.10 :Đánh giá PSNR của khung hình tổng hợp giữa các phương pháp truyền thống và thuật toán Hole filling SWA – (a): Chuỗi Pantomime; (b): Chuỗi Champagne; (c): Chuỗi Balloons; (d): Chuỗi Kendo; (e): Chuỗi Newsletter; (f): Chuỗi Lovebird

KẾT LUẬN

Luận văn đã trình bày một phương pháp Hole filling SWA bao gồm tiền xử lý xóa bỏ các nhiễu biên được sử dụng cho tổng hợp khung hình ảo. Nhiễu biên xảy ra do ánh xạ lỗi giữa ảnh độ sâu và ảnh vân trong suốt quá trình tổng hợp. Sau khi loại bỏ các nhiễu biên. Để lấp đầy các hố, luận văn đã sử dụng thuật toán trọng số trung bình đường xoắn ốc và kỹ thuật tìm kiếm gradient. Thuật toán trọng số trung bình theo đường xoắn ốc giữ biên của đối tượng tốt bằng cách sử dụng thông tin về độ sâu và thuật toán tìm kiếm gradient giữ được các thông tin chi tiết. Luận văn đã kết hợp những điểm mạnh của cả hai thuật toán.

TÀI LIỆU THAM KHẢO

[1] M. Tanimoto, “Targets of MPEG FTV” FTV Seminar, July 2014

[2] https://en.wikipedia.org/wiki/Free_viewpoint_television

[3] “Proposal on a New Activity for the Third Phase of FTV” ISO/IEC JTC1/SC29/WG11 MPEG2012/M30229, July 2013, Vienna, Austria.

[4] http://www.epixea.com/research/multi-view-coding-thesisse18.html

[5] https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding

[6] Min Soo Ko* and Jisang Yoo “Virtual View Generation by a New Hole Filling Algorithm”, 2014, J Electr Eng Technol Vol. 9

[7] https://en.wikipedia.org/wiki/Inpainting

[8] http://www.fujii.nuee.nagoya-u.ac.jp/multiview-data/mpeg/mpeg_ftv.html

[9] F. Dufaux, B. Pesquet-Popescu, M. Cagnazzo, “Emerging Technologies for 3D Video: Creation, Coding, Transmission and Rendering”

[10] https://en.wikipedia.org/wiki/Time-of-flight_camera

[11] “Depth estimation reference software (DERS) 5.0 “, M Tanimoto, T Fujii, K Suzuki, N Fukushima, Y Mori - ISO/IEC JTC1/SC29/WG11 M, 2009

[12] https://en.wikipedia.org/wiki/Computer_stereo_vision

[13] W. SUN, L. XU, Oscar C. AU, S. H. CHUI, C. W. KWOK, “An overview of free viewpoint Depth-Image-Based Rendering (DIBR)”, Proceedings of the APSIPA, Singapore, December 2010

[14] Tian D, Lai P, Lopez P, Gomila C, "View synthesis techniques for 3D video.", Proceedings applications of digital image processing XXXII, vol 7443, pp 74430T– 1–11, 2009