

**ĐẠI HỌC QUỐC GIA HÀ NỘI**  
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**



**VŨ DUY KHƯƠNG**

**PHÁT TRIỂN THUẬT TOÁN NỘI SUY NHẪM 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 CẢM 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 của 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

|   |    |
|---|----|
| <b>LỜI CAM ĐOAN</b> .....   | 2  |
| <b>LỜI CẢM ƠN</b> .....   | 3  |
| <b>MỤC LỤC</b> .....  | 4  |
| <b>DANH MỤC KÝ HIỆU, TỪ VIẾT TẮT</b> .....  | 6  |
| <b>DANH MỤC HÌNH VẼ</b> .....   | 7  |
| <b>DANH MỤC BẢNG BIỂU</b> .....   | 9  |
| <b>MỞ ĐẦU</b> .....   | 10 |
| <b>CHƯƠNG 1: ĐẶT VẤN ĐỀ</b> .....   | 13 |
| 1.1. Lý do chọn đề tài.....   | 13 |
| 1.2. Mục tiêu của luận văn.....   | 13 |
| 1.3. Cấu trúc luận văn.....   | 13 |
| <b>CHƯƠNG 2: CÁC KHÁI NIỆM CƠ BẢN</b> .....                                       | 15 |
| 2.1. Các ứng dụng video giả lập 3D.....   | 15 |
| 2.1.1. Tivi 3D (3DTV).....  | 15 |
| 2.1.2. Tivi Free Viewpoint (FTV).....   | 16 |
| 2.2. Các định dạng biểu diễn video 3D.....  | 17 |
| 2.2.1. Video đa khung hình (MVV) và Video đa khung hình với độ sâu<br>(MVVD)..... | 18 |
| 2.2.2. Bản đồ độ sâu.....   | 20 |
| 2.3. Biểu diễn dựa trên bản đồ độ sâu (DIBR).....                                 | 23 |
| 2.3.1. Tổng hợp 3D.....   | 23 |
| 2.3.2. Sáp nhập khung hình.....   | 27 |
| 2.3.3. Hole filling các vùng Disocclusions.....                                   | 28 |
| 2.4. Phần mềm tham chiếu tổng hợp khung hình (VSRS).....                          | 30 |

|   |           |
|---|-----------|
| 2.4.1. Trạng thái tổng quát.....                          | 30        |
| 2.4.2. Trạng thái 1D.....                                 | 32        |
| 2.5. Thuật toán tổng hợp khung hình Fast 1-D.....         | 33        |
| 2.5.1. Chuẩn hóa mẫu.....                                 | 35        |
| 2.5.2. Tổng hợp, nội suy và hole filling.....             | 35        |
| 2.5.3. Tạo bản đồ xác thực.....                           | 37        |
| 2.5.4. Tăng cường sự đồng nhất.....                       | 37        |
| 2.5.5. Kết hợp.....                                       | 38        |
| <b>CHƯƠNG 3: THUẬT TOÁN HOLE FILLING SWA.....</b>         | <b>39</b> |
| 3.1. Giới thiệu thuật toán Hole filling SWA.....          | 39        |
| 3.2. Thuật toán Hole filling SWA.....                     | 39        |
| 3.2.1. Phát hiện nhiễu biên.....                          | 39        |
| 3.2.2. Xác định thứ tự Hole filling đối với vùng nền..... | 42        |
| 3.2.3. Thuật toán trọng số trung bình đường xoắn ốc.....  | 43        |
| 3.2.4. Thuật toán tìm kiếm Gradient.....                  | 45        |
| <b>CHƯƠNG 4: CÀI ĐẶT VÀ KẾT QUẢ THỰC NGHIỆM.....</b>      | <b>46</b> |
| 4.1. Cài đặt thực nghiệm.....                             | 46        |
| 4.2. Kết quả tổng hợp khung hình.....                     | 48        |
| <b>KẾT LUẬN.....</b>                                      | <b>57</b> |
| <b>TÀI LIỆU THAM KHẢO.....</b>                            | <b>58</b> |

## DANH MỤC KÝ HIỆU, TỪ VIẾT TẮT

| <b>Số</b> | <b>Thuật Ngữ</b> | <b>Giải Thích</b>                 |
|-----------|------------------|-----------------------------------|
| 1         | TV               | Television                        |
| 2         | 3D               | Three Dimension                   |
| 3         | MVD              | Multiview Video plus Depth        |
| 4         | MVV              | Multi View 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

| Số        | Tên Hình   | Trang |
|-----------|--|-------|
| Hình 2.1  | Minh họa nguyên lý nhìn của con người  | 16    |
| Hình 2.2  | Hệ thống FTV tổng quát   | 17    |
| Hình 2.3  | Ví dụ về một cảnh biểu diễn video đa khung hình  | 18    |
| Hình 2.4  | Ví dụ về sắp xếp một hệ thống camera đa khung hình   | 19    |
| Hình 2.5  | Ví dụ về video đa khung hình với chiều sâu   | 20    |
| Hình 2.6  | Một khung màu và bản đồ độ sâu liên quan   | 20    |
| Hình 2.7  | Công thức tính độ lệch   | 22    |
| Hình 2.8  | Framework khung hình tổng hợp cơ bản sử dụng 2 camera đầu vào                                  | 23    |
| Hình 2.9  | Chuyển đổi hệ tọa độ thực sang hệ tọa độ camera  | 24    |
| Hình 2.10 | Cấu trúc hình học của camera pin-hole (a) 3D và (b) 2D   | 24    |
| Hình 2.11 | Tổng hợp khung hình với hai khung hình dữ liệu MVD   | 26    |
| Hình 2.12 | Cấu hình lập thể, tất cả điểm ảnh không nhìn thấy từ các điểm quan sát camera                  | 29    |
| Hình 2.13 | Phương pháp hole filling truyền thống  | 30    |
| Hình 2.14 | Biểu đồ luồng dữ liệu của phần mềm VSRS trạng thái tổng quát                                   | 31    |
| Hình 2.15 | Biểu đồ luồng phần mềm VSRS 1D mode  | 33    |
| Hình 2.16 | Thuật toán tổng hợp khung hình   | 34    |
| Hình 2.17 | Sự phụ thuộc giữa các tín hiệu đầu vào, trung gian và đầu ra của bước tính toán lỗi, biểu diễn | 36    |
| Hình 3.1  | Nhiều biên   | 40    |
| Hình 3.2  | Các hồ chung   | 40    |
| Hình 3.3  | Sơ đồ khối thuật toán Hole filling SWA   | 41    |
| Hình 3.4  | Thuật toán Hole filling SWA loại bỏ nhiễu biên   | 42    |
| Hình 3.5  | (a) Thứ tự thuật toán Hole filling SWA; (b) Kết quả  | 42    |
| Hình 3.6  | Biểu đồ luồng thuật toán trọng số trung bình đường xoắn ốc                                     | 44    |

|           |  |    |
|-----------|--|----|
| Hình 3.7  | Thuật toán tìm kiếm Gradient, bước (1) và bước (2)   | 45 |
| Hình 4.1  | File cấu hình chương trình .cfg  | 47 |
| Hình 4.2  | Giao diện chạy chương trình  | 47 |
| Hình 4.3  | Tổng hợp khung hình trong trường hợp nội suy   | 48 |
| Hình 4.4  | Khung hình ảo tổng hợp – “Balloons”  | 49 |
| Hình 4.5  | Khung hình ảo tổng hợp – “Champagne”   | 49 |
| Hình 4.6  | Khung hình ảo tổng hợp – “Kendo”   | 50 |
| Hình 4.7  | Khung hình ảo tổng hợp – “Pantomime”   | 51 |
| Hình 4.8  | Khung hình ảo tổng hợp - “Lovebird”  | 51 |
| Hình 4.9  | Khung hình ảo tổng hợp - “Newspaper”   | 52 |
| Hình 4.10 | Đánh giá PSNR của khung hình tổng hợp giữa các phương pháp truyền thống và thuật toán Hole filling SWA | 56 |



## DANH MỤC BẢNG BIỂU

| <b>Số</b> | <b>Tên Bảng</b>   | <b>Trang</b> |
|-----------|---|--------------|
| Bảng 4.1  | Các chuỗi được sử dụng trong thí nghiệm                       | 46           |
| Bảng 4.2  | So sánh hiệu năng PSNR giữa các thuật toán trong các phần mềm | 54           |

## 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

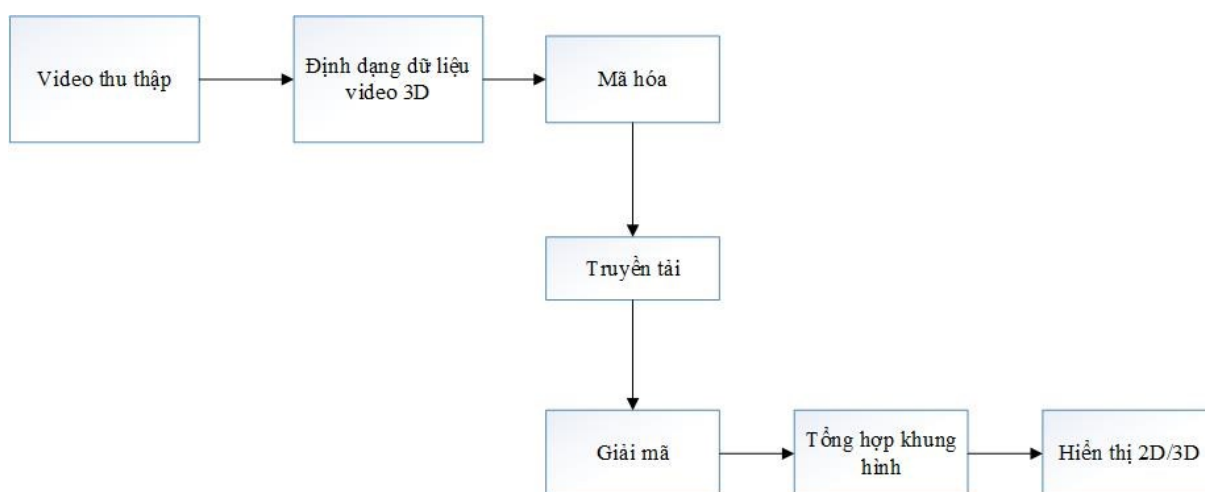
### 2.1. CÁC ỨNG DỤNG VIDEO GIẢ LẬP 3D

#### 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 biệt 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 gọi 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.

#### 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



**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. 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 .

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. 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

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.

### 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 .

Độ lệch của một đối tượng trong hệ tọa độ camera có thể được suy ra một cách dễ dàng từ công thức sau:



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

ở đây:

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

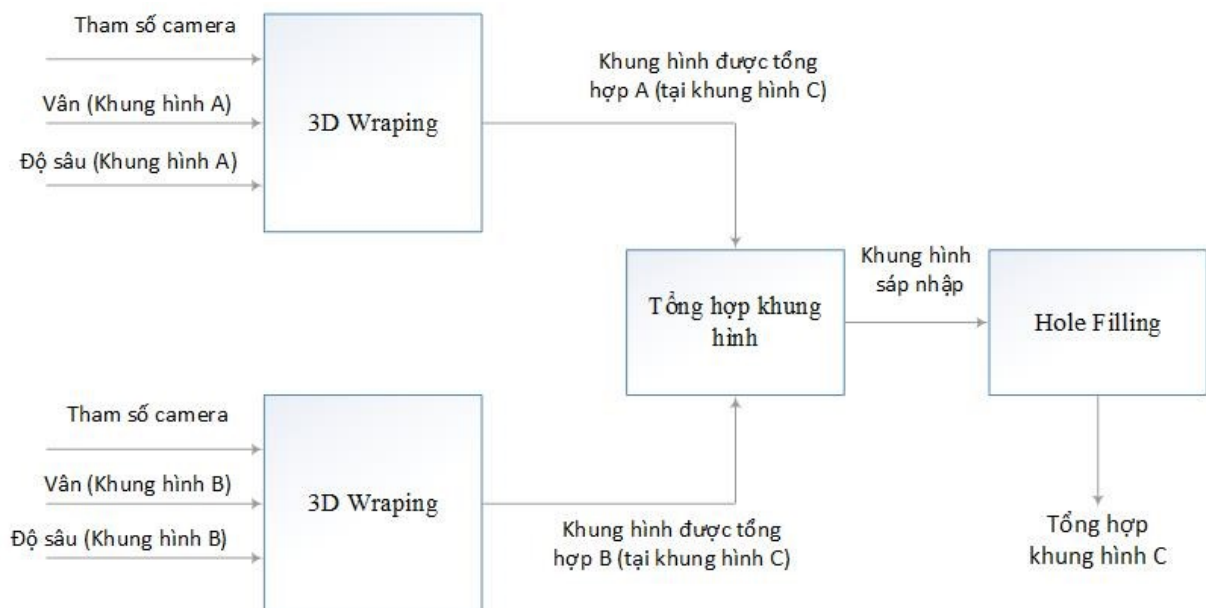
$f$  là chiều dài tiêu cự.  $T$  là khoảng cách giữa camera (cơ bản).

$Z$  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. .

### 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  $M$  ( $M \geq 1$ ) 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



**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 Wrapping) 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.

### **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.

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.

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.

### **2.3.3. HOLE FILLING CÁC VÙNG DISOCCLUSIONS**

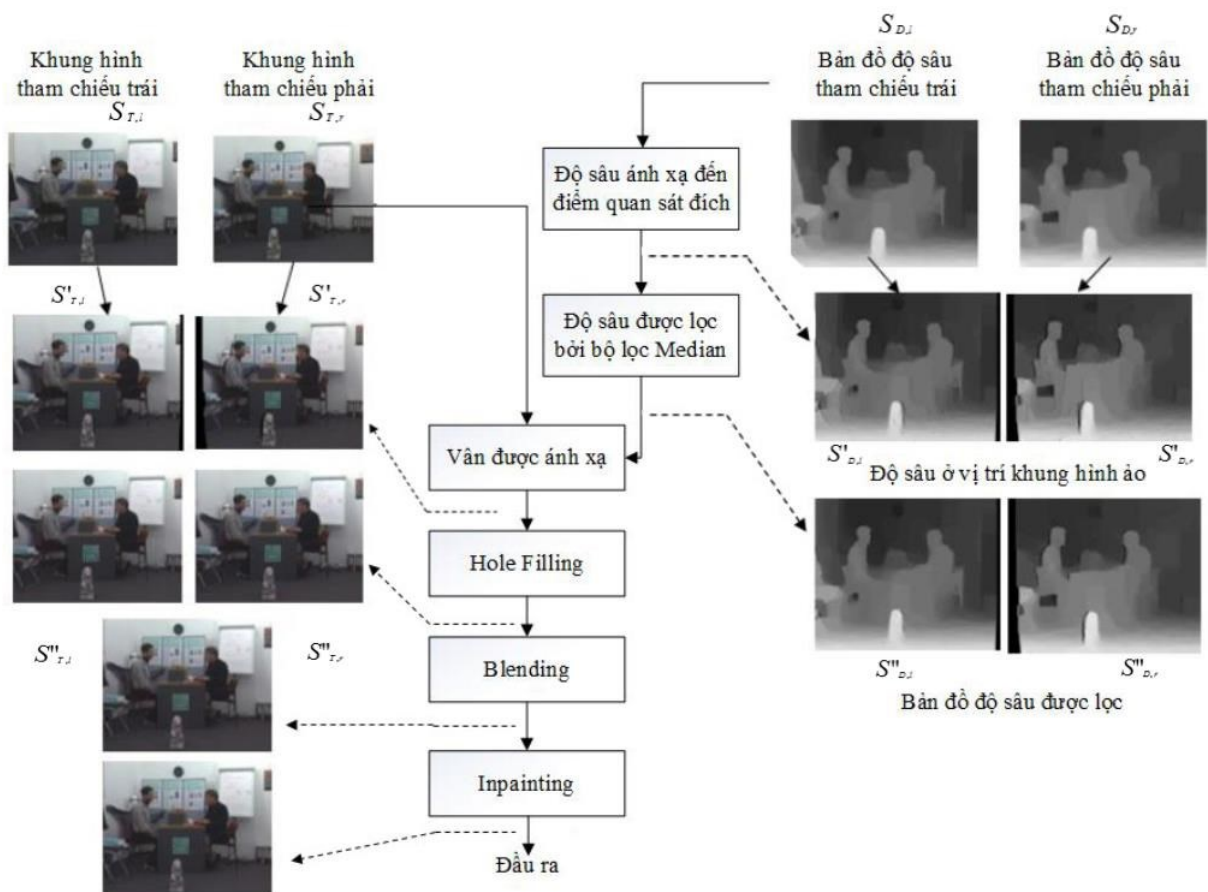
Để cung cấp cho người xem trải nghiệm hoàn thiện, các hố trong khung hình biểu diễn cần được loại bỏ. Có hai hướng chính để giải quyết vấn đề này. Một hướng là xử lý trước bản đồ độ sâu bằng cách làm mịn vùng không liên tục của bản đồ độ sâu trước khi dùng phương pháp DIBR loại bỏ vùng disocclusion trong 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

## **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.

### **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.



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

## 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 thiết 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.

## 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

### 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  $\frac{1}{4}$  theo chiều ngang và thành phần chroma được chuẩn hóa với phân số  $\frac{1}{8}$  theo chiều ngang và  $\frac{1}{2}$  theo chiều dọc. Mục đích của bước này là nhằm giảm đi các hồ 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.

### 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 disclusion 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.

### 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

### 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  $s'_{D,r}$ .

## **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 hõ 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 hõ 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 hõ 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 hõ, 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.

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.

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

### 3.2.2. XÁC ĐỊNH THUẬT 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

### 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.

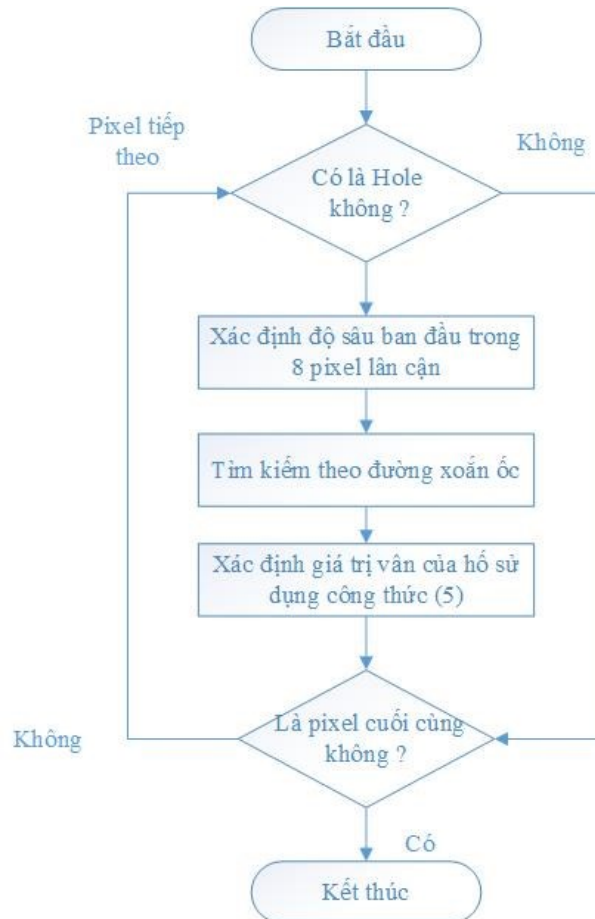
(2) Các điểm ảnh hố ban đầu được chọn trong bước (1) được gán với giá trị độ sâu nhỏ nhất của 8 giá trị lân cận mà không phụ thuộc vào vùng hố chung. Thực hiện tìm kiếm theo đường xoắn ốc ở vị trí hố chung ban đầu theo thứ tự tìm kiếm

(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

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

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

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



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

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

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

- (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 chọn 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

| <b>Độ phân giải<br/>Rộng*Cao</b> | <b>Chuỗi</b> | <b>Số lượng<br/>khung hình</b> | <b>Các khung<br/>hình</b> |
|----------------------------------|--------------|--------------------------------|---------------------------|
| 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

```

C:\Windows\System32\cmd.exe - TappRenderer.exe -c rendering_2view_balloons_holefilling_mode2.cfg
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

E:\TONG HOP 3D>TappRenderer.exe -c rendering_2view_balloons_holefilling_mode2.cfg

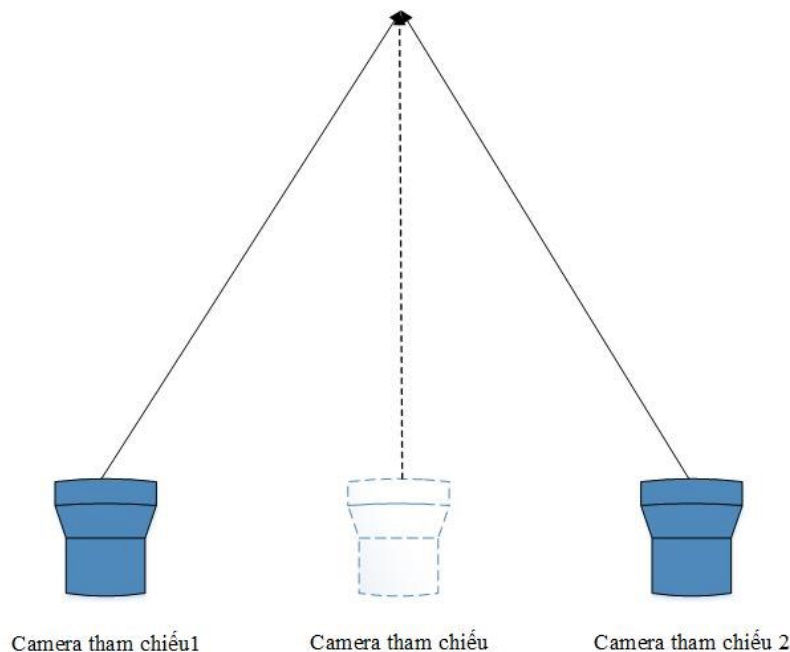
Interpolating camera parameters for virtual view(s): 2
InputVideoFile_0 : E:\TONG HOP 3D\balloons1.yuv
InputVideoFile_1 : E:\TONG HOP 3D\balloons3.yuv
InputDepthFile_0 : E:\TONG HOP 3D\depth_balloons_1.yuv
InputDepthFile_1 : E:\TONG HOP 3D\depth_balloons_3.yuv
SynthOutputFile_0 : E:\TONG HOP 3D\hevc_virtual_balloons2_holefilling_mode2.yuv
Format : 1024x768
Frame index : 0 - 299 (300 frames)
CameraParameterFile : E:\TONG HOP 3D\cam_balloons.cfg
BaseViewNumbers : 1 3 (2 views)
Sweep : 0
SynthViewNumbers : 50000 (1 views)
Log2SamplingFactor : 0
UVUp : 1
PreProcMode : 0
PreFilterSize : 0
SimEnhance : 1
BlendMode : 0
BlendZThresPerc : 30
BlendUseDistWeight : 1
BlendHoleMargin : 6
InterpolationMode : 4
HolefillingMode : 2
PostProcMode : 0
ShiftPrecision : 2
TemporalDepthFilter : 0
RenderMode : 0
RendererDirection : 0

Rendering Frame 0 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 1 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 2 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 3 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 4 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 5 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 6 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0
Rendering Frame 7 of View 2 Left BaseView: 1 Right BaseView: 3 BlendMode: 0

```

Hình 4.2 : Giao diện chạy chương trình

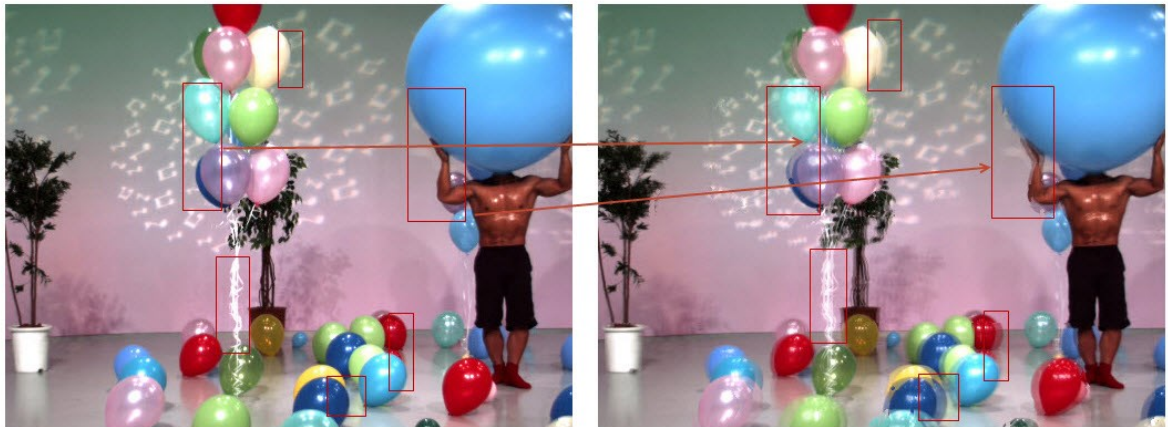
## 4.2. KẾT QUẢ TỔNG HỢP KHUNG HÌNH



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

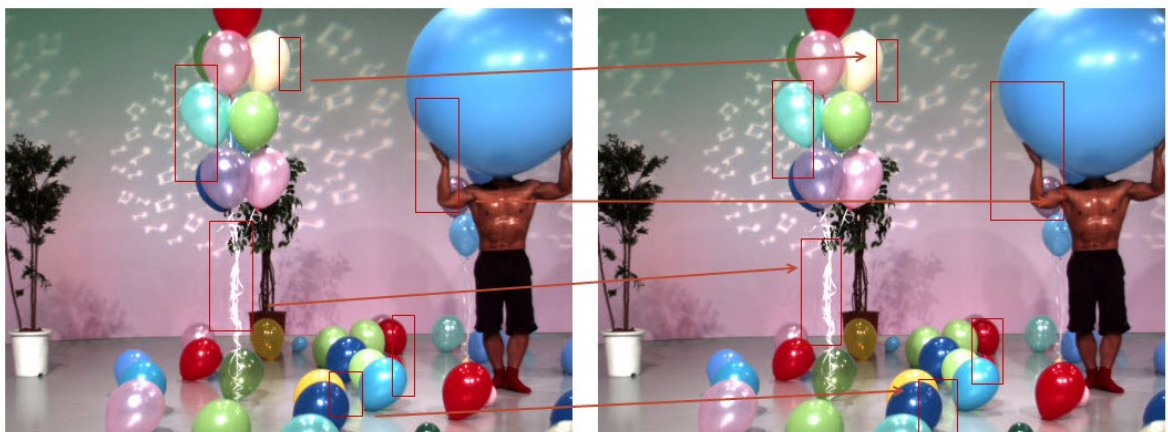
Trong Hình 4.1 chỉ ra trường hợp tổng hợp khung hình nội suy. Kết quả sinh ra một hình ảnh điểm quan sát ảo ở vị trí phía trong của các khung hình tham chiếu bằng thuật toán Hole filling SWA và so sánh hiệu năng của nó so với thuật toán Hole filling trong VSRS 4.0 alpha; thuật toán Hole filling trong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



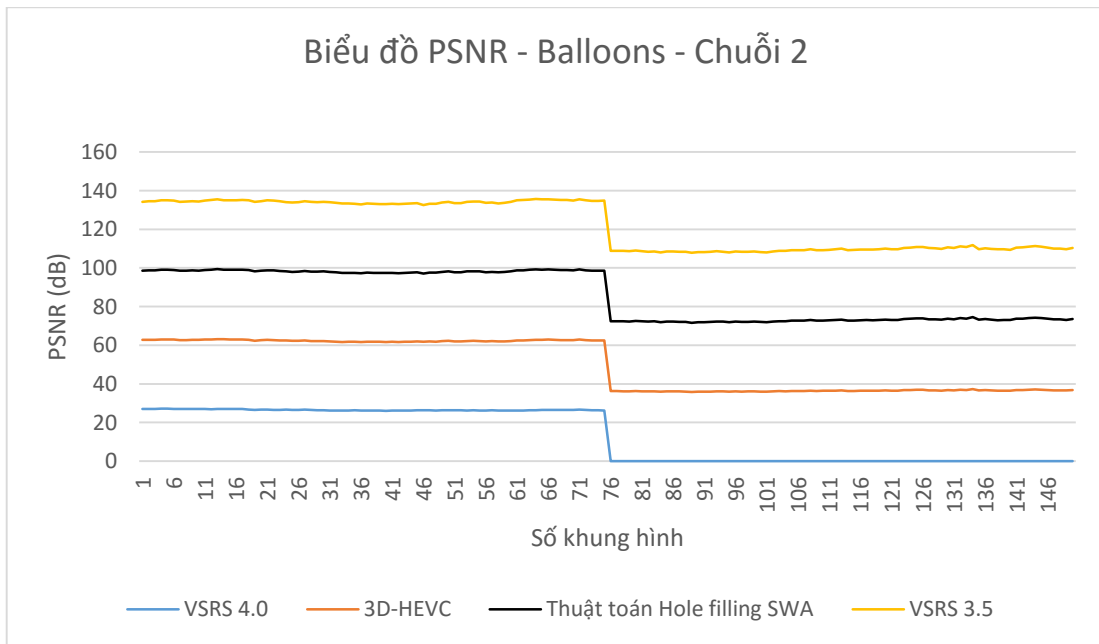
(c)

(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

| Tệp 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



**Hình 4.10 :**Đánh giá PSNR của khung hình tổng hợp giữa các phương pháp truyền thống và thuật toán Hole filling SWA – Chuỗi Balloons

## 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 hõ, 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](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](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.nuce.nagoya-u.ac.jp/multiview-data/mpeg/mpeg\\_ftv.html](http://www.fujii.nuce.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](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](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