**CHƯƠNG 11: MPEG Video Coding:**

**MPEG-1,2,4 và 7**

**11.1. Tổng quan:**

**-** Nhóm Chuyên gia Hình ảnh Động (MPEG) được thành lập vào năm 1988 nhằm tạo ra một tiêu chuẩn cho việc truyền tải video và âm thanh số. Số lượng thành viên đã tăng nhanh chóng từ khoảng 25 chuyên gia vào năm 1988 lên thành một cộng đồng bao gồm hàng trăm công ty và tổ chức [1]. Điều quan trọng là phải bảo đảm các lợi ích sở hữu trí tuệ được duy trì trong hệ thống tiêu chuẩn MPEG. Điều này được thực hiện bằng cách chỉ định nghĩa một luồng bit nén mà từ đó ngầm định cấu trúc của bộ giải mã. Các thuật toán nén, và do đó cả các bộ mã hóa, hoàn toàn phụ thuộc vào các nhà sản xuất.

- Trong chương này, chúng ta sẽ nghiên cứu một số vấn đề thiết kế quan trọng nhất của MPEG-1 và MPEG-2, tiếp theo là những kiến thức cơ bản về các tiêu chuẩn sau này, MPEG-4 và MPEG-7, với những mục tiêu rất khác biệt.

- Với sự xuất hiện của các tiêu chuẩn nén video mới như H.264 và H.265 (sẽ được thảo luận trong Chương 12), có thể nhiều người sẽ xem các tiêu chuẩn MPEG này là cũ, tức là lỗi thời. Tuy nhiên, điều này không phải là vấn đề vì:

(a) Công nghệ cơ bản của mã hóa lai và các khái niệm quan trọng nhất mà chúng ta sẽ giới thiệu ở đây, chẳng hạn như bù chuyển động, mã hóa biến đổi dựa trên DCT, và khả năng mở rộng, đều được sử dụng trong tất cả các tiêu chuẩn cũ và mới.

(b) Mặc dù cách tiếp cận biểu diễn và nén video dựa trên đối tượng hình ảnh được phát triển trong MPEG-4 và MPEG-7 chưa được sử dụng phổ biến trong các tiêu chuẩn hiện nay, nhưng nó có tiềm năng lớn để được áp dụng trong tương lai khi công nghệ Thị giác Máy tính cần thiết cho việc phát hiện đối tượng tự động trở nên dễ dàng tiếp cận hơn.

**11.2. MPEG-1:**

**-** Tiêu chuẩn nén âm thanh/video số MPEG-1 [2,3] đã được nhóm MPEG của Tổ chức Tiêu chuẩn Quốc tế/Tổ chức Kỹ thuật Điện Quốc tế (ISO/IEC) phê duyệt vào tháng 11 năm 1991 để mã hóa Hình ảnh Động và Âm thanh Liên quan cho phương tiện lưu trữ số với tốc độ lên đến khoảng 1.5 Mbit/s [4]. Các phương tiện lưu trữ số phổ biến bao gồm đĩa compact (CD) và đĩa video compact (VCD). Trong tổng số 1.5 Mbit/s quy định, 1.2 Mbps được dành cho video đã mã hóa và 256 kbps (kilobit mỗi giây) có thể được sử dụng cho âm thanh stereo. Điều này mang lại chất lượng hình ảnh tương đương với băng VHS và chất lượng âm thanh ngang bằng với âm thanh CD.

- Nhìn chung, MPEG-1 áp dụng định dạng TV số CCIR601, còn được gọi là Định dạng Nguồn Đầu vào (SIF). MPEG-1 chỉ hỗ trợ video không xen kẽ. Thông thường, độ phân giải hình ảnh của nó là 352 × 240 cho video NTSC ở 30 khung hình mỗi giây (fps) hoặc 352 × 288 cho video PAL ở 25 fps. Nó sử dụng phương pháp lấy mẫu màu 4:2:0.

Tiêu chuẩn MPEG-1, còn được gọi là ISO/IEC 11172 [4], bao gồm năm phần:

* 11172-1 Hệ thống,
* 11172-2 Video,
* 11172-3 Âm thanh,
* 11172-4 Tuân thủ,
* và 11172-5 Phần mềm.

- Tóm tắt, Hệ thống xử lý nhiều vấn đề, bao gồm chia đầu ra thành các gói luồng bit, ghép kênh và đồng bộ hóa các luồng video và âm thanh. Tuân thủ (hoặc sự phù hợp) chỉ định thiết kế của các bài kiểm tra để xác minh liệu một luồng bit hoặc bộ giải mã có tuân thủ tiêu chuẩn hay không. Phần mềm bao gồm một triển khai phần mềm đầy đủ của bộ giải mã tiêu chuẩn MPEG-1 và một triển khai phần mềm mẫu của bộ mã hóa.

- Giống như trong H.261 và H.263, MPEG-1 sử dụng công nghệ Mã hóa Lai, tức là kết hợp giữa dự đoán chuyển động giữa các khung hình và mã hóa biến đổi trên các lỗi dư. Chúng ta sẽ xem xét các tính năng chính của mã hóa video MPEG-1 và sẽ để thảo luận về mã hóa âm thanh MPEG cho Chương 14.

**11.2.1. Bù Đắp Chuyển Động trong MPEG-1:**

- Như đã thảo luận trong chương trước, mã hóa video dựa trên bù đắp chuyển động trong H.261 hoạt động như sau: Trong quá trình ước lượng chuyển động (motion estimation), mỗi macroblock của khung P-frame mục tiêu được gán với một macroblock phù hợp nhất từ khung I hoặc P đã được mã hóa trước đó. Điều này được gọi là dự đoán (prediction). Sự khác biệt giữa macroblock và macroblock phù hợp của nó được gọi là lỗi dự đoán (prediction error), và nó được gửi đến DCT (Biến đổi Cosin rời rạc) và các bước mã hóa tiếp theo.

( **Macroblock** (khối vĩ mô) là một đơn vị cơ bản trong mã hóa video, đặc biệt trong các chuẩn nén video như H.261, MPEG, và H.264. Một macroblock thường bao gồm một tập hợp các pixel và có thể được sử dụng để nén thông tin hình ảnh hiệu quả hơn. )

- Vì dự đoán được thực hiện từ một khung hình trước, nên nó được gọi là **dự đoán tiến (forward prediction)**. Do sự chuyển động bất ngờ và sự che khuất trong các cảnh thực tế, macroblock mục tiêu có thể không tìm thấy một khối phù hợp tốt trong khung hình trước. Hình 11.1 minh họa rằng macroblock chứa một phần của quả bóng trong khung hình mục tiêu không thể tìm thấy một macroblock phù hợp tốt trong khung hình trước, bởi vì một nửa quả bóng đã bị che khuất bởi một đối tượng khác. Tuy nhiên, một sự phù hợp có thể dễ dàng được tìm thấy từ khung hình tiếp theo.

- MPEG giới thiệu một loại khung thứ ba —B-frames (khung B)— cùng với bù đắp chuyển động hai chiều (bidirectional motion compensation) đi kèm. Hình 11.2 minh họa ý tưởng mã hóa B-frame dựa trên bù đắp chuyển động. Ngoài dự đoán tiến (forward prediction), một dự đoán lùi (backward prediction) cũng được thực hiện, trong đó macroblock phù hợp được lấy từ một khung I hoặc P tương lai trong chuỗi video. Do đó, mỗi macroblock từ một B-frame sẽ chỉ định tối đa hai vector chuyển động: một từ dự đoán tiến và một từ dự đoán lùi.

A diagram of a target and a target frame

Description automatically generated

A diagram of a diagram

Description automatically generated

- Nếu việc tìm kiếm khớp ở cả hai hướng thành công, hai vector chuyển động sẽ được gửi đi, và hai macroblock phù hợp tương ứng sẽ được trung bình (được chỉ định bằng ký hiệu “%” trong hình) trước khi so sánh với macroblock mục tiêu để tạo ra lỗi dự đoán. Nếu chỉ có thể tìm thấy một khớp chấp nhận được trong một trong hai khung tham chiếu, chỉ một vector chuyển động và macroblock tương ứng sẽ được sử dụng từ dự đoán tiến hoặc dự đoán lùi.

- Hình 11.3 minh họa một chuỗi khung video có thể xảy ra. Mẫu khung thực tế được xác định tại thời điểm mã hóa và được chỉ định trong tiêu đề của video. MPEG sử dụng ký hiệu M để chỉ khoảng cách giữa một P-frame và khung I hoặc P-frame trước đó, và ký hiệu N để chỉ khoảng cách giữa hai khung I liên tiếp. Trong Hình 11.3, M = 3 và N = 9. Một trường hợp đặc biệt là M = 1, khi không sử dụng B-frame.

A diagram of a sequence of steps

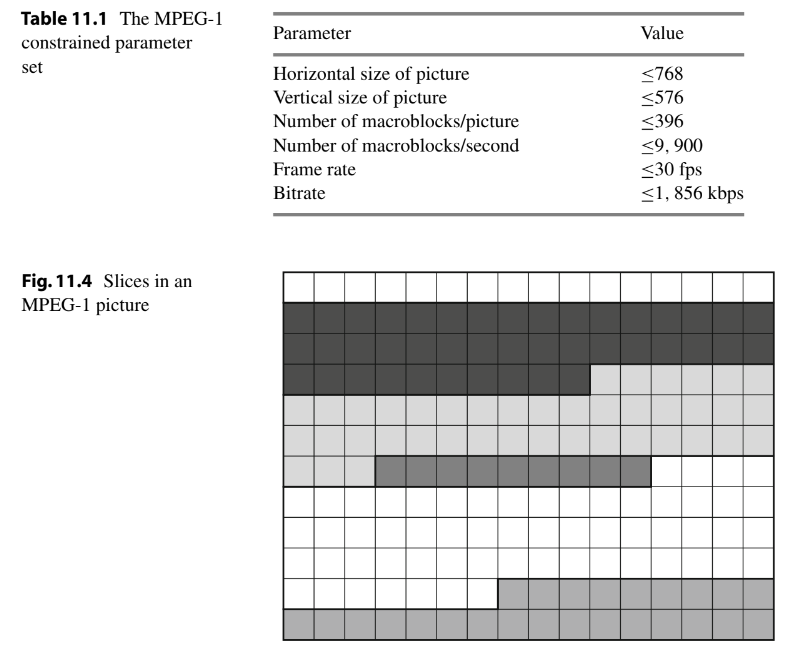
Description automatically generated

- Vì bộ mã hóa và giải mã MPEG không thể làm việc với bất kỳ macroblock nào từ một B-frame mà không có khung P hoặc I kế tiếp, nên thứ tự mã hóa và truyền tải thực tế (được hiển thị ở dưới cùng của Hình 11.3) khác với thứ tự hiển thị của video (được hiển thị ở trên). Sự chậm trễ không thể tránh khỏi và nhu cầu về bộ đệm trở thành một vấn đề quan trọng trong truyền tải mạng thời gian thực, đặc biệt là trong việc phát video MPEG trực tuyến.

**11.2.2. Các điểm khác biệt chính khác so với H.261:**

- Ngoài việc giới thiệu bù đắp chuyển động hai chiều (các B-frame), MPEG-1 cũng khác với H.261 ở những khía cạnh sau:

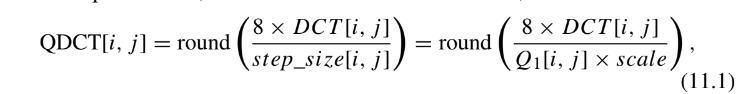
* **Định dạng nguồn**: H.261 chỉ hỗ trợ các định dạng nguồn CIF (352 × 288) và QCIF (176 × 144). MPEG-1 hỗ trợ định dạng SIF (352×240 cho NTSC, 352×288 cho PAL). Nó cũng cho phép chỉ định các định dạng khác, miễn là các tham số ràng buộc (CPS), được hiển thị trong Bảng 11.1, được đáp ứng.
* **Slices**: Thay vì GOB như trong H.261, một bức tranh MPEG-1 có thể được chia thành một hoặc nhiều slices (Hình 11.4), mà linh hoạt hơn so với GOB. Chúng có thể chứa số lượng macroblock khác nhau trong một bức tranh duy nhất và có thể bắt đầu và kết thúc ở bất kỳ đâu, miễn là chúng lấp đầy toàn bộ bức tranh. Mỗi slice được mã hóa độc lập. Ví dụ, các slices có thể có các hệ số tỷ lệ khác nhau trong bộ lượng hóa. Điều này cung cấp thêm tính linh hoạt trong việc kiểm soát bitrate. Hơn nữa, khái niệm slice là quan trọng cho việc phục hồi lỗi, vì mỗi slice có một mã slice\_start\_code duy nhất. Một slice trong MPEG tương tự như GOB trong H.261 (và H.263): nó là cấp độ thấp nhất trong cấu trúc phân cấp của MPEG có thể được phục hồi hoàn toàn mà không cần giải mã toàn bộ bộ mã biến độ dài trong dòng bit.
* **Lượng hóa**: Lượng hóa MPEG-1 sử dụng các bảng lượng hóa khác nhau cho mã hóa trong (intra) và giữa (inter) (Bảng 11.2 và 11.3). Các số lượng hóa cho mã hóa trong (Bảng 11.2) thay đổi trong một macroblock. Điều này khác với H.261, trong đó tất cả các số lượng hóa cho các hệ số AC là không đổi trong một macroblock.



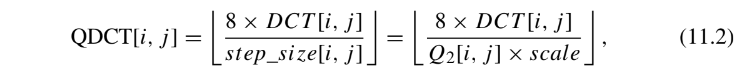
A table with numbers and a few words

Description automatically generated with medium confidence

- Giá trị **step\_size[i, j]** hiện được xác định bởi tích của **Q[i, j]** và **scale**, trong đó **Q1** hoặc **Q2** là một trong các bảng lượng hóa ở trên và **scale** là một số nguyên trong khoảng [1, 31]. Sử dụng **DCT** và **QDCT** để ký hiệu các hệ số DCT trước và sau khi lượng hóa, cho các hệ số DCT trong chế độ mã hóa trong (intramode),



và cho các hệ số DCT trong chế độ mã hóa giữa (intermode),



với **Q1** và **Q2** lần lượt đề cập đến Bảng 11.2 và Bảng 11.3. Một lần nữa, một toán tử làm tròn thường được sử dụng trong Phương trình (11.1) và do đó không để lại vùng chết nào, trong khi một toán tử làm tròn xuống được sử dụng trong Phương trình (11.2), để lại một vùng chết trung tâm trong không gian lượng hóa của nó.

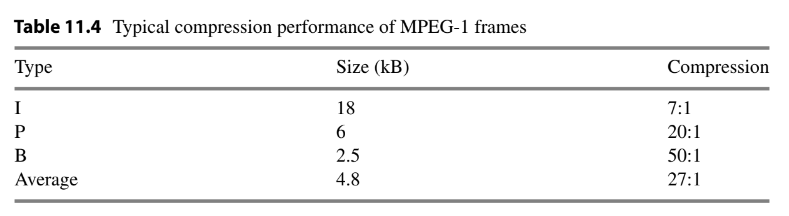
* Để tăng độ chính xác của các dự đoán dựa trên bù đắp chuyển động và do đó giảm thiểu lỗi dự đoán, MPEG-1 cho phép các vector chuyển động có độ chính xác dưới điểm ảnh (subpixel precision) (1/2 pixel). Kỹ thuật nội suy bilinear được thảo luận trong Mục 10.5.1 cho H.263 có thể được sử dụng để tạo ra các giá trị cần thiết tại các vị trí nửa điểm ảnh.
* MPEG-1 hỗ trợ khoảng cách lớn hơn giữa các khung I và P và do đó là một phạm vi tìm kiếm vector chuyển động lớn hơn nhiều. So với phạm vi tối đa ±15 pixel cho các vector chuyển động trong H.261, MPEG-1 hỗ trợ phạm vi [−512, 511.5] cho độ chính xác nửa pixel và [−1.024, 1.023] cho các vector chuyển động có độ chính xác đầy đủ. Tuy nhiên, do giới hạn thực tế trong độ phân giải hình ảnh, một phạm vi tối đa lớn như vậy có thể không bao giờ được sử dụng.
* Dòng bit MPEG-1 cho phép truy cập ngẫu nhiên. Điều này được thực hiện thông qua lớp Nhóm Hình ảnh (Group of Pictures - GOP), trong đó mỗi GOP được mã thời gian. Ngoài ra, khung đầu tiên trong bất kỳ GOP nào là một I-frame, điều này loại bỏ nhu cầu tham chiếu các khung khác. Do đó, lớp GOP cho phép bộ giải mã tìm kiếm một vị trí cụ thể trong dòng bit và bắt đầu giải mã từ đó.

- Bảng 11.4 liệt kê kích thước điển hình (tính bằng kilobytes) cho tất cả các loại khung MPEG-1. Có thể thấy rằng kích thước điển hình của các khung P đã nén nhỏ hơn đáng kể so với các khung I, vì nén giữa các khung khai thác tính dư thừa theo thời gian. Đặc biệt, các khung B còn nhỏ hơn các khung P, một phần do lợi thế của dự đoán hai chiều. Điều này cũng bởi vì các khung B thường được ưu tiên thấp nhất về mặt bảo tồn chất lượng; do đó, một tỷ lệ nén cao hơn có thể được áp dụng.

**11.2.3. Dòng Bit Video MPEG-1:**

- Hình 11.5 mô tả sáu lớp phân cấp cho dòng bit của video MPEG-1.

1. **Lớp Sequence**: Một chuỗi video bao gồm một hoặc nhiều nhóm hình ảnh (GOPs). Nó luôn bắt đầu với một tiêu đề chuỗi. Tiêu đề này chứa thông tin về hình ảnh, chẳng hạn như kích thước chiều ngang (**horizontal\_size**) và kích thước chiều dọc (**vertical\_size**), tỷ lệ pixel (**pixel\_aspect\_ratio**), tốc độ khung hình (**frame\_rate**), tốc độ bit (**bit\_rate**), kích thước bộ đệm (**buffer\_size**), ma trận lượng hóa (**quantization\_matrix**), và nhiều thông tin khác. Các tiêu đề chuỗi tùy chọn giữa các GOP có thể chỉ ra sự thay đổi về tham số.

****

**A diagram of a sequence

Description automatically generated**

1. **Lớp GOP**: Một GOP chứa một hoặc nhiều hình ảnh, trong đó phải có ít nhất một hình ảnh I (I-picture). Tiêu đề GOP chứa thông tin như **time\_code** để chỉ ra giờ-phút-giây-khung từ đầu chuỗi.
2. **Lớp Hình ảnh**: Ba loại hình ảnh phổ biến trong MPEG-1 là I-picture (mã hóa nội), P-picture (mã hóa dự đoán), và B-picture (mã hóa dự đoán hai chiều), như đã thảo luận ở trên. Cũng có một loại ít phổ biến hơn, D-picture (mã hóa DC), trong đó chỉ giữ lại các hệ số DC. MPEG-1 không cho phép trộn lẫn D-picture với các loại hình ảnh khác, điều này khiến D-picture trở nên không thực tế.
3. **Lớp Slice**: Như đã đề cập trước đó, MPEG-1 đã giới thiệu khái niệm slice để kiểm soát bitrate và để phục hồi và đồng bộ hóa sau khi mất hoặc bị hỏng dữ liệu. Các slice có thể có số lượng macroblock khác nhau trong một hình ảnh duy nhất. Độ dài và vị trí của mỗi slice được chỉ định trong tiêu đề.
4. **Lớp Macroblock**: Mỗi macroblock bao gồm bốn khối Y, một khối Cb và một khối Cr. Tất cả các khối đều có kích thước 8 × 8.
5. **Lớp Khối**: Nếu các khối được mã hóa nội, hệ số DC chênh lệch (DPCM của các hệ số DC, như trong JPEG) sẽ được gửi trước, tiếp theo là các mã biến độ dài (VLC) cho các hệ số AC. Ngược lại, cả hai hệ số DC và AC đều được mã hóa bằng các mã biến độ dài.

Mitchell và cộng sự [5] cung cấp thông tin chi tiết về các tiêu đề trong các lớp khác nhau của MPEG-1.

**11.3. MPEG-2:**

- Sự phát triển của tiêu chuẩn MPEG-2 bắt đầu vào năm 1990. Khác với MPEG-1, vốn chủ yếu là tiêu chuẩn để lưu trữ và phát video trên đĩa CD của một máy tính với tốc độ bit thấp (1,5 Mbps), MPEG-2 [6] được thiết kế cho video chất lượng cao hơn với tốc độ bit trên 4 Mbps. Nó ban đầu được phát triển như một tiêu chuẩn cho truyền hình số.

- Vào cuối những năm 1980, Truyền hình Cao cấp (Advanced TV - ATV) đã được hình dung để phát sóng HDTV qua các mạng mặt đất. Trong quá trình phát triển MPEG-2, ATV số cuối cùng đã chiếm ưu thế so với nhiều nỗ lực ban đầu về giải pháp analog cho HDTV. MPEG-2 đã đáp ứng được các yêu cầu về nén và tốc độ bit của truyền hình số/HDTV và thực tế đã thay thế một tiêu chuẩn riêng biệt, MPEG-3, mà ban đầu được cho là cần thiết cho HDTV.

- Tiêu chuẩn nén âm thanh/video MPEG-2, còn được gọi là ISO/IEC 13818 [7], đã được nhóm Chuyên gia Hình ảnh ISO/IEC phê duyệt vào tháng 11 năm 1994. Tương tự như MPEG-1, tiêu chuẩn này có các phần cho Hệ thống, Video, Âm thanh, Tuân thủ, và Phần mềm, cùng với các khía cạnh khác. Phần 2, phần nén video của tiêu chuẩn, ISO/IEC 13818-2, còn được biết đến với tên gọi H.262 trong ITU-T (Liên minh Viễn thông Quốc tế).

- MPEG-2 đã được chấp nhận rộng rãi không chỉ trong việc phát sóng truyền hình số qua các mạng mặt đất, vệ tinh hoặc cáp. Trong số các ứng dụng khác như Truyền hình Tương tác, nó cũng đã được áp dụng cho đĩa video kỹ thuật số hoặc đĩa đa phương tiện kỹ thuật số (DVD).

- MPEG-2 định nghĩa bảy hồ sơ (profiles) nhắm đến các ứng dụng khác nhau (ví dụ: hội nghị video độ trễ thấp, video có thể mở rộng, HDTV). Các hồ sơ này bao gồm: Simple, Main, SNR scalable, Spatially scalable, High, 4:2:2, và Multiview (trong đó hai cái nhìn sẽ đề cập đến video 3D). Trong mỗi hồ sơ, có tối đa bốn cấp độ (levels) được định nghĩa. Như Bảng 11.5 cho thấy, không phải tất cả các hồ sơ đều có bốn cấp độ. Ví dụ, hồ sơ Simple chỉ có cấp độ Main; trong khi hồ sơ High không có cấp độ Low.

- Bảng 11.6 liệt kê bốn cấp độ trong hồ sơ Main, với số lượng dữ liệu tối đa và các ứng dụng mục tiêu. Ví dụ, cấp độ High hỗ trợ độ phân giải hình ảnh cao 1.920 × 1.152, tốc độ khung hình tối đa là 60 fps, tốc độ pixel tối đa là 62,7 × 10^6 pixel mỗi giây, và tốc độ dữ liệu tối đa sau khi mã hóa là 80 Mbps. Cấp độ Low được nhắm đến video SIF; do đó, nó cung cấp khả năng tương thích ngược với MPEG-1. Cấp độ Main dành cho video CCIR601, trong khi các cấp độ High 1440 và High được nhắm đến HDTV châu Âu và HDTV Bắc Mỹ, tương ứng.

- Thông số kỹ thuật video DVD chỉ cho phép bốn độ phân giải hiển thị: chẳng hạn như, ở 29,97 fps, kiểu interlaced, 720 × 480, 704 × 480, 352 × 480, và 352 × 240. Do đó, tiêu chuẩn video DVD chỉ sử dụng một hình thức hạn chế của hồ sơ Main MPEG-2 tại các cấp độ Main và Low.

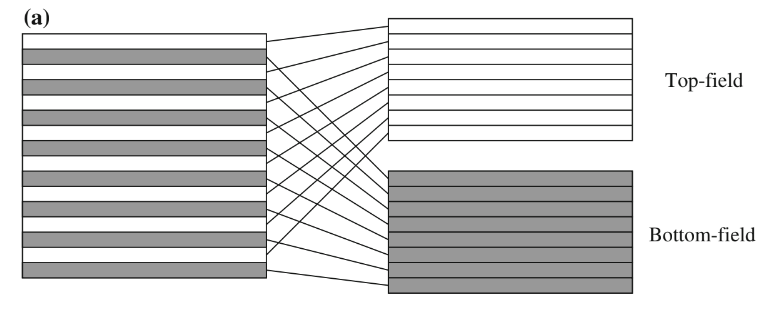
A table of information

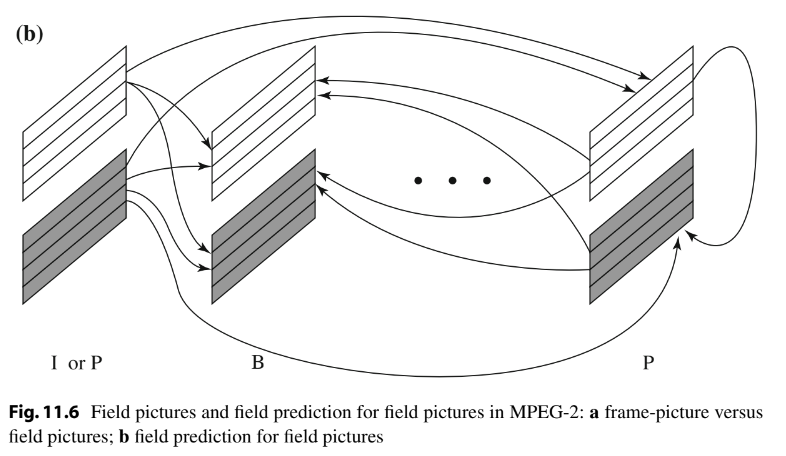
Description automatically generated with medium confidence

**11.3.1. Hỗ trợ Video Interlaced:**

- MPEG-1 chỉ hỗ trợ video không phân tách (quét tiến). Vì MPEG-2 được áp dụng cho truyền hình số, nó cũng phải hỗ trợ video phân tách, vì đây là một trong những tùy chọn cho truyền hình số và HDTV.

- Như đã đề cập trước đó, trong video phân tách, mỗi khung hình bao gồm hai trường, được gọi là trường trên (top-field) và trường dưới (bottom-field). Trong một khung hình (frame-picture), tất cả các dòng quét từ cả hai trường được chèn lẫn nhau để tạo thành một khung hình duy nhất. Khung hình này sau đó được chia thành các macroblock 16 × 16 và được mã hóa bằng cách sử dụng bù đắp chuyển động. Mặt khác, nếu mỗi trường được coi như một hình ảnh riêng biệt, thì nó được gọi là hình ảnh trường (field picture). Như Hình 11.6a cho thấy, mỗi khung hình có thể được chia thành hai hình ảnh trường. Hình này cho thấy 16 dòng quét từ một khung hình ở bên trái, so với 8 dòng quét trong mỗi phần của hai trường ở bên phải.



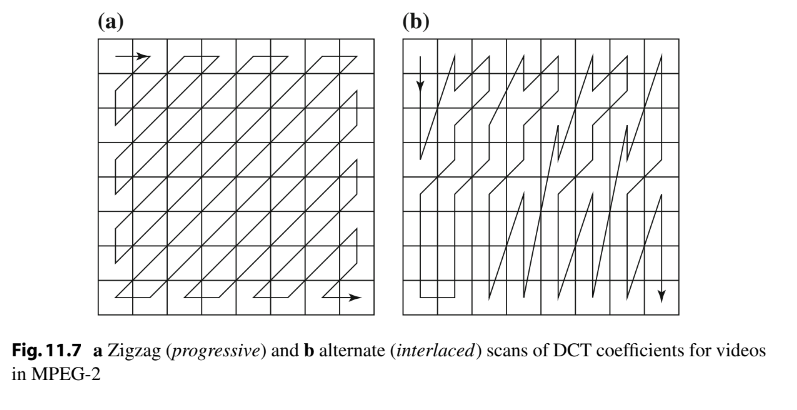


- Chúng ta thấy rằng, về mặt diện tích hiển thị trên màn hình/tivi, mỗi macroblock 16 cột × 16 hàng trong hình ảnh trường tương ứng với một khu vực 16 × 32 trong hình ảnh khung, trong khi mỗi macroblock 16 × 16 trong hình ảnh khung tương ứng với một khu vực 16 × 8 trong hình ảnh trường. Như được trình bày dưới đây, quan sát này sẽ trở thành một yếu tố quan trọng trong việc phát triển các chế độ dự đoán khác nhau cho mã hóa video dựa trên bù đắp chuyển động.

**Năm Chế Độ Dự Đoán**

MPEG-2 định nghĩa dự đoán khung hình và dự đoán trường, cũng như năm chế độ dự đoán khác nhau, phù hợp với nhiều ứng dụng khác nhau, nơi yêu cầu về độ chính xác và tốc độ của bù đắp chuyển động có sự khác biệt.

* **Dự đoán khung hình cho hình ảnh khung**. Chế độ này tương tự như các phương pháp dự đoán dựa trên bù đắp chuyển động trong MPEG-1 cho cả P-frame và B-frame. Dự đoán khung hình hoạt động hiệu quả cho các video chỉ có chuyển động của đối tượng và camera chậm hoặc ở mức độ vừa phải.
* **Dự đoán trường cho hình ảnh trường**. (Xem Hình 11.6b) Chế độ này sử dụng kích thước macroblock là 16×16 từ hình ảnh trường. Đối với hình ảnh P-field (những hình ở phía bên phải trong hình), dự đoán được thực hiện dựa trên hai trường đã được mã hóa gần đây nhất. Các macroblock trong hình ảnh trường trên được dự đoán từ các hình ảnh trường trên hoặc trường dưới của khung I hoặc P trước đó. Các macroblock trong hình ảnh trường dưới được dự đoán từ hình ảnh trường trên của cùng một khung hoặc hình ảnh trường dưới của khung I hoặc P trước đó. Đối với hình ảnh B-field, dự đoán cả hai chiều tiến và lùi được thực hiện từ các hình ảnh trường của các khung I hoặc P trước và sau. Không có quy định nào yêu cầu duy trì sự đồng bộ giữa các trường—có nghĩa là, các hình ảnh trường trên và dưới có thể được dự đoán từ bất kỳ trường nào trong các hình ảnh tham chiếu.
* **Dự đoán trường cho hình ảnh khung**. Chế độ này xử lý trường trên và trường dưới của một hình ảnh khung một cách riêng biệt. Do đó, mỗi macroblock 16 × 16 từ hình ảnh khung mục tiêu được chia thành hai phần 16×8, mỗi phần đến từ một trường. Dự đoán trường được thực hiện cho các phần 16 × 8 này theo cách tương tự như mô tả trong Hình 11.6b. Ngoài kích thước khối nhỏ hơn, sự khác biệt duy nhất là trường dưới sẽ không được dự đoán từ trường trên của cùng một khung, vì chúng ta đang xử lý hình ảnh khung. Ví dụ, đối với hình ảnh P-frame, phần dưới 16×8 sẽ được dự đoán từ một trong hai trường của khung I hoặc P trước đó. Do đó, hai vector chuyển động sẽ được tạo ra cho mỗi macroblock 16 × 16 trong hình ảnh P-frame. Tương tự, có thể tạo ra tối đa bốn vector chuyển động cho mỗi macroblock trong hình ảnh B-frame.
* **Bù đắp chuyển động 16 × 8 cho hình ảnh trường**. Mỗi macroblock 16 × 16 từ hình ảnh trường mục tiêu giờ đây được chia thành hai nửa 16 × 8—tức là, tám dòng đầu tiên và tám dòng tiếp theo. Dự đoán trường được thực hiện cho mỗi nửa. Kết quả là, hai vector chuyển động sẽ được tạo ra cho mỗi macroblock 16 × 16 trong hình ảnh P-field và tối đa bốn vector chuyển động cho mỗi macroblock trong hình ảnh B-field. Chế độ này rất hiệu quả trong việc bù đắp chuyển động một cách chính xác hơn khi chuyển động diễn ra nhanh chóng và không đồng nhất.
* **Dual-prime cho P-pictures**. Đây là chế độ duy nhất có thể được sử dụng cho cả hình ảnh khung và hình ảnh trường. Ban đầu, dự đoán trường được thực hiện từ mỗi trường trước đó có cùng độ chẵn (trên hoặc dưới). Mỗi vector chuyển động MV sau đó được sử dụng để suy ra một vector chuyển động tính toán CV trong trường có độ chẵn ngược lại, xem xét tỷ lệ thời gian và độ dịch chuyển theo chiều dọc giữa các dòng trong các trường trên và dưới. Bằng cách này, cặp MV và CV tạo ra hai dự đoán sơ bộ cho mỗi macroblock. Lỗi dự đoán của chúng được tính trung bình và được sử dụng làm lỗi dự đoán cuối cùng. Chế độ này nhằm mô phỏng dự đoán B-frame cho P-frame mà không áp dụng dự đoán lùi (do đó giảm độ trễ trong mã hóa).



**Quét thay thế và Field\_DCT**

- Quét thay thế và Field\_DCT là các kỹ thuật nhằm cải thiện hiệu quả của DCT trên các lỗi dự đoán. Chúng chỉ áp dụng cho các hình ảnh khung trong video interlaced.

- Sau khi thực hiện dự đoán khung hình trong các hình ảnh khung, lỗi dự đoán được gửi đến DCT, trong đó mỗi khối có kích thước 8 × 8. Do đặc điểm của video interlaced, các dòng liên tiếp trong các khối này đến từ các trường khác nhau; do đó, có ít sự tương quan giữa chúng hơn so với các dòng thay thế. Điều này cho thấy rằng các hệ số DCT ở tần số không gian dọc thấp có xu hướng có độ lớn giảm so với các hệ số trong video không interlaced.

- Dựa trên phân tích trên, một quét thay thế được giới thiệu. Nó có thể được áp dụng theo từng hình ảnh trong MPEG-2 như một lựa chọn thay thế cho quét zigzag. Như Hình 11.7a cho thấy, quét zigzag giả định rằng trong video không interlaced, các hệ số DCT ở góc trên bên trái của khối thường có độ lớn lớn hơn. Quét thay thế (Hình 11.7b) nhận ra rằng trong video interlaced, các thành phần tần số không gian cao hơn về mặt dọc có thể có độ lớn lớn hơn và do đó cho phép chúng được quét sớm hơn trong chuỗi. Các thí nghiệm đã chỉ ra rằng quét thay thế có thể cải thiện PSNR lên tới 0,3 dB so với quét zigzag và hiệu quả nhất cho các video có chuyển động nhanh.

- Trong MPEG-2, Field\_DCT có thể giải quyết vấn đề tương tự. Trước khi áp dụng DCT, các dòng trong macroblock của các hình ảnh khung có thể được sắp xếp lại, sao cho tám dòng đầu tiên đến từ trường trên và tám dòng cuối cùng đến từ trường dưới. Điều này khôi phục sự dư thừa (và tương quan) không gian cao hơn giữa các dòng liên tiếp. Việc sắp xếp lại sẽ được đảo ngược sau IDCT. Field\_DCT không áp dụng cho hình ảnh màu, nơi mỗi macroblock chỉ có 8 × 8 pixel.

**11.3.2. Tính năng mở rộng của MPEG-2:**

- Giống như trong JPEG2000, khả năng mở rộng cũng là một vấn đề quan trọng đối với MPEG-2. Vì MPEG-2 được thiết kế cho nhiều ứng dụng khác nhau, bao gồm TV kỹ thuật số và HDTV, video sẽ thường xuyên được truyền qua các mạng có đặc điểm rất khác nhau. Do đó, cần có một luồng bit mã hóa đơn lẻ có khả năng mở rộng cho các tốc độ bit khác nhau.

- Mã hóa mở rộng của MPEG-2 còn được gọi là mã hóa phân lớp, trong đó có thể định nghĩa một lớp cơ bản và một hoặc nhiều lớp nâng cao. Lớp cơ bản có thể được mã hóa, truyền và giải mã độc lập để có được chất lượng video cơ bản. Việc mã hóa và giải mã lớp nâng cao, tuy nhiên, phụ thuộc vào lớp cơ bản hoặc lớp nâng cao trước đó. Thường chỉ sử dụng một lớp nâng cao, gọi là mã hóa mở rộng hai lớp.

- Mã hóa mở rộng phù hợp với video MPEG-2 được truyền qua các mạng có các đặc điểm sau:

* **Tốc độ bit rất khác nhau:** Nếu tốc độ liên kết chậm, chỉ luồng bit từ lớp cơ bản sẽ được gửi. Ngược lại, nếu tốc độ cao hơn, các luồng bit từ một hoặc nhiều lớp nâng cao cũng sẽ được gửi để đạt được chất lượng video cải thiện.
* **Kênh tốc độ bit biến đổi (VBR):** Khi tốc độ bit của kênh giảm, luồng bit từ ít hoặc không có lớp nâng cao sẽ được truyền, và ngược lại.
* **Kết nối có nhiễu:** Lớp cơ bản có thể được bảo vệ tốt hơn hoặc gửi qua các kênh ít nhiễu hơn.

- Hơn nữa, mã hóa mở rộng rất lý tưởng cho việc truyền phát tiến bộ: các luồng bit từ lớp cơ bản được gửi trước, để người dùng có thể nhanh chóng có cái nhìn cơ bản về video, sau đó dữ liệu và chất lượng dần dần được cải thiện. Điều này hữu ích cho việc cung cấp TV kỹ thuật số tương thích (ATV) và HDTV.

- MPEG-2 hỗ trợ các khả năng mở rộng sau:

* **Mở rộng SNR:** Lớp nâng cao cung cấp tỷ lệ tín hiệu trên nhiễu cao hơn.
* **Mở rộng không gian:** Lớp nâng cao cung cấp độ phân giải không gian cao hơn.
* **Mở rộng thời gian:** Lớp nâng cao cho phép tốc độ khung hình cao hơn.
* **Mở rộng kết hợp:** Kết hợp bất kỳ hai trong ba khả năng mở rộng trên.
* **Phân vùng dữ liệu:** Các hệ số DCT được lượng tử hóa được chia thành các phân vùng.

**Tỷ lệ SNR**

- Hình 11.8 minh họa cách thức hoạt động của tỷ lệ SNR trong bộ mã hóa và giải mã MPEG-2.

- Bộ mã hóa SNR của MPEG-2 tạo ra hai luồng bit đầu ra: **Bits\_base** và **Bits\_enhance** ở hai lớp khác nhau. Ở lớp cơ bản, các hệ số DCT được lượng tử hóa thô, dẫn đến việc sử dụng ít bit hơn và chất lượng video tương đối thấp. Sau khi mã hóa theo độ dài biến đổi, luồng bit này được gọi là **Bits\_base**.Các hệ số DCT được lượng tử hóa thô sau đó được giải lượng tử hóa ngược (**Q⁻¹**) và đưa vào lớp nâng cao để so sánh với hệ số DCT gốc.

A diagram of a machine

Description automatically generated

A diagram of a decoder

Description automatically generated

- Sự khác biệt giữa chúng được lượng tử hóa tinh để tạo ra một hệ số DCT tinh chỉnh, sau khi được mã hóa theo độ dài biến đổi, trở thành luồng bit được gọi là **Bits\_enhance**. Các hệ số DCT thô và tinh chỉnh được giải lượng tử hóa ngược sau đó được cộng lại, và sau khi thực hiện biến đổi ngược DCT (**IDCT**), chúng được sử dụng cho dự đoán bù chuyển động cho khung hình tiếp theo. Vì việc tinh chỉnh/nâng cao này so với lớp cơ bản cải thiện tỷ lệ tín hiệu trên nhiễu (SNR), nên loại khả năng mở rộng này được gọi là **khả năng mở rộng SNR**.

- Nếu vì lý do nào đó (ví dụ: sự cố kênh mạng), **Bits\_enhance** từ lớp nâng cao không thể thu được, thì sơ đồ mở rộng trên vẫn có thể hoạt động chỉ với **Bits\_base**. Trong trường hợp đó, đầu vào từ bộ giải lượng tử ngược (**Q⁻¹**) của lớp nâng cao chỉ cần được coi là bằng không. Bộ giải mã (Hình 11.8b) hoạt động ngược lại so với bộ mã hóa. Cả **Bits\_base** và **Bits\_enhance** đều được giải mã độ dài biến đổi (**VLD**) và giải lượng tử ngược (**Q⁻¹**) trước khi được cộng lại để khôi phục các hệ số DCT.

- Các bước còn lại giống như trong bất kỳ bộ giải mã video dựa trên bù chuyển động nào. Nếu cả hai luồng bit (**Bits\_base** và **Bits\_enhance**) đều được sử dụng, video đầu ra **Output\_high** sẽ có chất lượng nâng cao. Nếu chỉ sử dụng **Bits\_base**, video đầu ra **Output\_base** sẽ có chất lượng cơ bản.

**Độ phân giải không gian (Spatial Scalability)**

- Lớp cơ bản và lớp nâng cao trong khả năng mở rộng không gian của MPEG-2 không gắn kết chặt chẽ như trong khả năng mở rộng SNR; do đó, loại mở rộng này đơn giản hơn một chút. Chúng tôi sẽ không trình bày chi tiết cả bộ mã hóa và giải mã như trước mà chỉ giải thích quy trình mã hóa thông qua các sơ đồ cấp cao.

- Lớp cơ bản được thiết kế để tạo ra một luồng bit của các khung hình có độ phân giải giảm. Kết hợp với lớp nâng cao sẽ tạo ra các khung hình với độ phân giải ban đầu.

* **Mã hóa lớp cơ bản**: Dữ liệu video gốc được giảm mẫu không gian theo hệ số 2 và gửi đến bộ mã hóa lớp cơ bản. Sau các bước mã hóa thông thường như bù chuyển động, DCT trên lỗi dự đoán, lượng tử hóa và mã hóa entropy, luồng bit đầu ra được gọi là *Bits\_base*.
* **Mã hóa lớp nâng cao**: Khối macro được dự đoán từ lớp cơ bản sau đó được nội suy không gian để đạt độ phân giải 16 × 16. Khối này được kết hợp với khối macro được dự đoán thông thường từ chính lớp nâng cao để tạo thành khối macro dự đoán phục vụ cho bù chuyển động trong mã hóa lớp. Nội suy không gian ở đây sử dụng kỹ thuật nội suy bilinear như đã thảo luận trước đó.

- Việc kết hợp các khối macro sử dụng một bảng trọng số đơn giản, với giá trị trọng số *w* nằm trong khoảng [0, 1.0]. Nếu *w = 0*, khối macro dự đoán từ lớp cơ bản không được xem xét. Nếu *w = 1*, dự đoán hoàn toàn dựa trên lớp cơ bản. Thông thường, cả hai khối macro dự đoán được kết hợp tuyến tính, sử dụng trọng số *w* và *1 − w* tương ứng. Để đạt được lỗi dự đoán tối thiểu, các bộ mã hóa MPEG-2 có một bộ phân tích để chọn các giá trị *w* khác nhau từ bảng trọng số dựa trên từng khối macro.

**Khả năng mở rộng theo thời gian\**

- Mã hóa khả năng mở rộng theo thời gian có cả lớp cơ bản và lớp nâng cao của video ở tỷ lệ tạm thời giảm (tốc độ khung hình). Tốc độ khung hình giảm cho các lớp thường giống nhau, tuy nhiên, chúng cũng có thể khác nhau. Hình ảnh từ lớp cơ bản và lớp nâng cao (các lớp) có cùng độ phân giải không gian như video đầu vào. Khi kết hợp, chúng phục hồi video về tỷ lệ tạm thời ban đầu.

A diagram of a block diagram

Description automatically generated

- Hình 11.10 minh họa cách thực hiện khả năng mở rộng theo thời gian trong MPEG-2. Video đầu vào được phân tách theo thời gian thành hai phần, mỗi phần mang một nửa tốc độ khung hình gốc. Như trước đây, bộ mã hóa lớp cơ bản thực hiện các quy trình mã hóa đơn lớp bình thường cho video đầu vào của nó và tạo ra luồng bit đầu ra là Bits\_base.

A diagram of a block diagram

Description automatically generated

- Dự đoán các macroblock tương ứng tại lớp nâng cao có thể được thực hiện theo hai cách: Dự đoán bù chuyển động giữa các lớp hoặc kết hợp dự đoán bù chuyển động và dự đoán bù chuyển động giữa các lớp.

• **Dự đoán bù chuyển động giữa các lớp (Hình 11.10b)** Các macroblock của B-frame cho bù chuyển động tại lớp nâng cao được dự đoán từ các khung trước và sau (có thể là I-, P- hoặc B-) tại lớp cơ bản, nhằm tận dụng khả năng dư thừa giữa các lớp trong bù chuyển động.

**• Dự đoán bù chuyển động kết hợp và dự đoán bù chuyển động giữa các lớp (Hình 11.10c)** Phương pháp này kết hợp thêm các ưu điểm của dự đoán phía trước thông thường và dự đoán giữa các lớp đã đề cập ở trên. Các macroblock của B-frame tại lớp nâng cao được dự đoán phía trước từ khung trước đó ở cùng lớp của nó và được “dự đoán lùi” từ khung trước đó (hoặc, thay thế, khung kế tiếp) ở lớp cơ bản. Ở khung đầu tiên, P-frame tại lớp nâng cao chỉ thực hiện dự đoán phía trước từ I-frame ở lớp cơ bản.

**Tính năng mở rộng hỗn hợp (Hybrid Scalability)**

* Bất kỳ hai trong số ba loại mở rộng trên có thể được kết hợp để hình thành tính năng mở rộng hỗn hợp. Các sự kết hợp bao gồm:
* Mở rộng hỗn hợp không gian và tạm thời
* Mở rộng hỗn hợp SNR và không gian
* Mở rộng hỗn hợp SNR và tạm thời
* Thường thì một bộ mã hóa hỗn hợp ba lớp sẽ được áp dụng, bao gồm lớp cơ bản, lớp nâng cao 1 và lớp nâng cao 2.
* Ví dụ, đối với mở rộng hỗn hợp không gian và tạm thời, lớp cơ bản và lớp nâng cao 1 sẽ cung cấp khả năng mở rộng không gian, trong khi các lớp nâng cao 1 và 2 sẽ cung cấp khả năng mở rộng tạm thời, trong đó lớp nâng cao 1 thực sự hoạt động như một lớp cơ bản.
* Đối với bộ mã hóa, dữ liệu video đầu vào trước tiên được phân tách tạm thời thành hai luồng: một luồng cho lớp nâng cao 2; luồng còn lại cho lớp nâng cao 1 và lớp cơ bản (sau khi giảm độ phân giải không gian hơn nữa cho lớp cơ bản).
* Bộ mã hóa tạo ra ba luồng bit đầu ra: (a) Bits\_base từ lớp cơ bản, (b) Bits\_enhance1 được cải thiện về không gian từ lớp nâng cao 1, và (c) Bits\_enhance2 được cải thiện về không gian và tạm thời từ lớp nâng cao 2.
* Cách triển khai của hai loại mở rộng hỗn hợp còn lại tương tự và được để lại cho người học tự thực hành.

**Phân chia dữ liệu ( Data Partitioning )**

* Dòng video đã nén được chia thành hai phân vùng. Phân vùng cơ bản chứa các hệ số DCT tần số thấp, trong khi phân vùng tăng cường chứa các hệ số DCT tần số cao. Mặc dù các phân vùng này đôi khi cũng được gọi là các lớp (lớp cơ bản và lớp tăng cường), nhưng nghiêm ngặt mà nói, phân chia dữ liệu không thực hiện cùng một loại mã hóa theo lớp, vì một dòng dữ liệu video duy nhất chỉ đơn giản được chia ra và không phụ thuộc thêm vào phân vùng cơ bản để tạo ra phân vùng tăng cường. Tuy nhiên, phân chia dữ liệu có thể hữu ích cho việc truyền tải qua các kênh bị nhiễu và cho việc truyền tải theo kiểu tiến bộ.

**11.3.3. Những khác biệt chính so với MPEG-1:**

* + **Khả năng chống lỗi bit tốt hơn:** Vì video MPEG-2 thường được truyền tải qua nhiều mạng khác nhau, trong đó có một số mạng không ổn định và có tiếng ồn, lỗi bit là điều không thể tránh khỏi. Để đối phó với điều này, hệ thống MPEG-2 có hai loại luồng: Luồng Chương trình (Program) và Luồng Vận chuyển (Transport). Luồng Chương trình tương tự như luồng Hệ thống trong MPEG-1; do đó, nó cũng hỗ trợ khả năng tương thích ngược với MPEG-1. Luồng Vận chuyển nhằm mục đích cung cấp khả năng chống lỗi và khả năng bao gồm nhiều chương trình với các cơ sở thời gian độc lập trong một luồng duy nhất, cho việc đa hợp không đồng bộ và truyền tải qua mạng. Thay vì sử dụng các gói dài, có độ dài thay đổi, như trong MPEG-1 và trong luồng Chương trình MPEG-2, nó sử dụng các gói có độ dài cố định (188 byte). Nó cũng có cú pháp tiêu đề mới, để kiểm tra và sửa lỗi tốt hơn.
  + **Hỗ trợ các phương pháp mẫu màu 4:2:2 và 4:4:4:** Ngoài mẫu màu 4:2:0 như trong H.261 và MPEG-1, MPEG-2 còn cho phép các phương pháp mẫu màu 4:2:2 và 4:4:4, nhằm nâng cao chất lượng màu sắc. Như đã thảo luận trong Chương 5, mỗi bức tranh độ chói trong 4:2:2 được giảm mẫu theo chiều ngang với hệ số 2, trong khi 4:4:4 là một trường hợp đặc biệt, trong đó không có việc giảm mẫu màu nào diễn ra.
  + **Lượng hóa phi tuyến:** Việc lượng hóa trong MPEG-2 tương tự như trong MPEG-1. Kích thước bước (step\_size) cũng được xác định bởi tích của Q[i, j] và scale, trong đó Q là một trong các bảng lượng hóa mặc định cho mã hóa nội hoặc giữa. Hai loại quy mô được cho phép. Đối với loại đầu tiên, quy mô giống như trong MPEG-1, là một số nguyên trong khoảng [1, 31] và scalei = i. Đối với loại thứ hai, tuy nhiên, tồn tại một mối quan hệ phi tuyến—tức là, scalei ≡ α i. Giá trị quy mô thứ i có thể được tra cứu trong Bảng 11.7.
  + **Cấu trúc slice hạn chế hơn:** MPEG-1 cho phép các slice vượt qua ranh giới hàng macroblock. Do đó, toàn bộ bức tranh có thể là một slice duy nhất. Slices trong MPEG-2 phải bắt đầu và kết thúc trong cùng một hàng macroblock. Nói cách khác, cạnh trái của một bức tranh luôn bắt đầu một slice mới, và slice dài nhất trong MPEG-2 chỉ có thể chứa một hàng macroblock.
  + **Định dạng video linh hoạt hơn:** Theo tiêu chuẩn, kích thước bức tranh MPEG-2 có thể lớn tới 16 k×16 k pixel. Trong thực tế, MPEG-2 chủ yếu được sử dụng để hỗ trợ các độ phân giải bức tranh khác nhau như được định nghĩa bởi DVD, ATV và HDTV.
* Tương tự như H.261, H.263 và MPEG-1, MPEG-2 chỉ xác định cú pháp bitstream và bộ giải mã. Điều này để lại nhiều không gian cho việc cải tiến trong tương lai, đặc biệt là ở phía mã hóa. Cú pháp luồng video MPEG-2 phức tạp hơn so với MPEG-1, và các tài liệu tham khảo tốt cho nó có thể được tìm thấy trong [6,7].

**11.4. MPEG-4:**

**11.4.1. Tổng quan về MPEG-4:**

- MPEG-1 và MPEG-2 sử dụng kỹ thuật mã hóa dựa trên khung hình, trong đó mỗi khung video hình chữ nhật được chia thành các macroblock và sau đó là các khối để nén. Đây cũng được gọi là mã hóa dựa trên khối. Mối quan tâm chính của họ là tỉ lệ nén cao và chất lượng video chấp nhận được dưới những kỹ thuật nén như vậy. MPEG-4 lại có một sự nhấn mạnh rất khác. Ngoài việc nén, MPEG-4 còn chú trọng đến sự tương tác của người dùng. Điều này cho phép một số lượng lớn người dùng tạo ra và truyền tải các bài trình bày và ứng dụng đa phương tiện của họ trên các cơ sở hạ tầng mới, chẳng hạn như Internet, World Wide Web (WWW), và mạng di động/kỹ thuật số. MPEG-4 đã đi xa hơn những người tiền nhiệm của nó bằng cách áp dụng một phương pháp mã hóa dựa trên đối tượng mới—các đối tượng đa phương tiện giờ đây là thực thể cho mã hóa MPEG-4. Các đối tượng đa phương tiện (còn được gọi là đối tượng âm thanh và hình ảnh) có thể là tự nhiên hoặc tổng hợp; nghĩa là, chúng có thể được ghi lại bằng camera video hoặc được tạo ra bởi các chương trình máy tính.

A diagram of a fruit processing process

Description automatically generated

- Mã hóa dựa trên đối tượng không chỉ có tiềm năng cung cấp tỉ lệ nén cao hơn mà còn có lợi cho việc biên tập, thao tác, lập chỉ mục và truy xuất video kỹ thuật số. Hình 11.11 minh họa cách mà video MPEG-4 có thể được biên tập và thao tác thông qua các thao tác đơn giản như chèn/xóa, dịch/chuyển, thay đổi kích thước, v.v., trên các đối tượng hình ảnh.

- MPEG-4 (phiên bản 1) đã được hoàn thiện vào tháng 10 năm 1998 và trở thành tiêu chuẩn quốc tế vào đầu năm 1999, được gọi là ISO/IEC 14496. Một phiên bản cải tiến (phiên bản 2) đã được hoàn thiện vào tháng 12 năm 1999 và đạt được tình trạng Tiêu chuẩn Quốc tế vào năm 2000. Tương tự như các tiêu chuẩn MPEG trước đó, năm phần đầu tiên của nó bao gồm Hệ thống, Video, Âm thanh, Tuân thủ và Phần mềm. Chương này sẽ thảo luận về các vấn đề nén video trong MPEG-4 Phần 2 (chính thức là ISO/IEC 14496-2).

- Ban đầu được nhắm đến cho các ứng dụng truyền thông băng thông thấp (từ 4.8–64 kbps cho các ứng dụng di động và lên đến 2 Mbps cho các ứng dụng khác), băng thông cho video MPEG-4 hiện nay bao trùm một phạm vi rộng, từ 5 kbps đến 10 Mbps.

- Như mô hình tham chiếu trong Hình 11.12a cho thấy, một hệ thống MPEG-1 chỉ đơn thuần cung cấp dữ liệu âm thanh và video từ bộ nhớ của nó và không cho phép bất kỳ sự tương tác nào từ người dùng. MPEG-2 đã thêm một thành phần Tương tác (được chỉ định bằng các đường chấm trong Hình 11.12a) và do đó cho phép những tương tác hạn chế từ người dùng trong các ứng dụng như video qua mạng và truyền hình tương tác. MPEG-4 (Hình 11.12b) là một tiêu chuẩn hoàn toàn mới cho (a) việc biên tập các đối tượng đa phương tiện để tạo ra các cảnh âm thanh-hình ảnh mong muốn, (b) phân multiplex và đồng bộ hóa các luồng bit cho các thực thể dữ liệu đa phương tiện này để chúng có thể được truyền tải với chất lượng dịch vụ (QoS) đảm bảo, và (c) tương tác với cảnh âm thanh-hình ảnh ở đầu nhận. MPEG-4 cung cấp một bộ công cụ các mô-đun mã hóa tiên tiến và các thuật toán cho việc nén âm thanh và video.

A diagram of a process

Description automatically generated

- MPEG-4 định nghĩa Định dạng Nhị phân cho Cảnh (BInary Format for Scenes - BIFS) [10], giúp việc biên tập các đối tượng đa phương tiện thành một cảnh. BIFS thường được thể hiện qua một đồ thị cảnh, trong đó các nút mô tả các nguyên tắc âm thanh-hình ảnh và thuộc tính của chúng, và cấu trúc đồ thị cho phép mô tả các mối quan hệ không gian và thời gian của các đối tượng trong cảnh. BIFS là một cải tiến của Ngôn ngữ Mô hình Thực tế Ảo (Virtual Reality Modeling Language - VRML). Đặc biệt, nó nhấn mạnh vào thời gian và đồng bộ hóa của các đối tượng, điều mà thiết kế VRML ban đầu còn thiếu. Ngoài BIFS, MPEG-4 (phiên bản 2) cung cấp một môi trường lập trình, MPEG-J [11], trong đó các ứng dụng Java (gọi là MPEGlets) có thể truy cập các gói và API của Java để tăng cường tính tương tác của người dùng cuối.

- Cấu trúc phân cấp của các luồng bit hình ảnh MPEG-4 rất khác với MPEG-1 và MPEG-2 ở chỗ nó tập trung nhiều vào các đối tượng video. Hình 11.13 minh họa năm cấp độ mô tả phân cấp của một cảnh trong các luồng bit hình ảnh MPEG-4. Nói chung, mỗi Dãy Đối tượng Video (Video-object Sequence - VS) sẽ có một hoặc nhiều Đối tượng Video (Video Objects - VOs), mỗi VO sẽ có một hoặc nhiều Lớp Đối tượng Video (Video Object Layers - VOLs), và cứ như vậy. Về mặt cú pháp, tất cả năm cấp độ đều có một mã bắt đầu duy nhất trong luồng bit, để cho phép truy cập ngẫu nhiên.

A screenshot of a video description

Description automatically generated

1. **Dãy Đối tượng Video (Video-object Sequence - VS)**: VS cung cấp toàn bộ cảnh hình ảnh MPEG-4, có thể chứa các đối tượng tự nhiên hoặc tổng hợp 2D hoặc 3D.
2. **Đối tượng Video (Video Object - VO)**: VO là một đối tượng cụ thể trong cảnh, có thể có hình dạng tùy ý (không hình chữ nhật), tương ứng với một đối tượng hoặc nền của cảnh.
3. **Lớp Đối tượng Video (Video Object Layer - VOL)**: VOL hỗ trợ việc mã hóa có khả năng mở rộng (đa lớp). Một VO có thể có nhiều VOL dưới mã hóa có khả năng mở rộng hoặc một VOL duy nhất dưới mã hóa không có khả năng mở rộng. Như một trường hợp đặc biệt, MPEG-4 cũng hỗ trợ một loại VOL đặc biệt với tiêu đề ngắn hơn. Điều này cung cấp khả năng tương thích luồng bit với H.263 cơ bản [12].
4. **Nhóm Các Mặt phẳng Đối tượng Video (Group of Video Object Planes - GOV)**: GOV nhóm các mặt phẳng đối tượng video. Đây là một cấp độ tùy chọn.
5. **Mặt phẳng Đối tượng Video (Video Object Plane - VOP)**: VOP là một bức ảnh của một VO tại một thời điểm cụ thể, phản ánh hình dạng, kết cấu và các tham số chuyển động của VO tại thời điểm đó. Nói chung, một VOP là một hình ảnh có hình dạng tùy ý. Một trường hợp suy biến trong mã hóa video MPEG-4 xảy ra khi toàn bộ khung hình video hình chữ nhật được coi là một VOP. Trong trường hợp này, nó tương đương với MPEG-1 và 2. MPEG-4 cho phép các VOP chồng lấp lên nhau, nghĩa là một VOP có thể che khuất một phần VOP khác trong một cảnh.

**11.4.2. Mã hóa Dựa trên Đối tượng Video trong MPEG-4:**

- MPEG-4 mã hóa/giải mã từng VOP một cách riêng biệt (thay vì xem xét toàn bộ khung hình). Do đó, mã hóa dựa trên đối tượng video của nó còn được gọi là mã hóa dựa trên VOP. Cuộc thảo luận của chúng tôi sẽ bắt đầu với mã hóa cho các đối tượng tự nhiên (thông tin chi tiết hơn có thể được tìm thấy trong [13,14]). Phần 11.4.3 mô tả mã hóa đối tượng tổng hợp.

**Mã hóa VOP so với mã hóa khung**

- MPEG-1 và MPEG-2 không hỗ trợ khái niệm VOP; do đó, phương pháp mã hóa của chúng được gọi là mã hóa khung. Bởi vì mỗi khung hình được chia thành nhiều macroblock, từ đó thực hiện mã hóa dựa trên bù chuyển động, nó cũng được gọi là mã hóa theo khối. Hình 11.14a cho thấy ba khung hình từ một chuỗi video với một chiếc xe đang di chuyển về phía trái và một người đi bộ đang đi theo hướng ngược lại. Hình 11.14b cho thấy mã hóa theo khối điển hình, trong đó vector chuyển động (MV) được thu được cho một trong các macroblock.

- Mã hóa video MPEG-1 và MPEG-2 chỉ quan tâm đến tỷ lệ nén và không xem xét sự tồn tại của các đối tượng hình ảnh. Do đó, các vector chuyển động được tạo ra có thể không nhất quán với chuyển động ở cấp độ đối tượng và sẽ không hữu ích cho phân tích và lập chỉ mục video dựa trên đối tượng.

- Hình 11.14c minh họa một ví dụ khả thi trong đó cả hai đối tượng tiềm năng đều tạo ra các lỗi dự đoán nhỏ. Nếu Đối Tượng Tiềm Năng 2 tạo ra một lỗi dự đoán (hơi) nhỏ hơn so với Đối Tượng Tiềm Năng 1, thì MV2 sẽ được chọn làm vector chuyển động cho macroblock trong phương pháp mã hóa theo khối, mặc dù chỉ có MV1 là nhất quán với hướng di chuyển của phương tiện.

- Mã hóa dựa trên đối tượng trong MPEG-4 nhằm giải quyết vấn đề này, bên cạnh việc cải thiện tỷ lệ nén. Hình 11.14d cho thấy mỗi VOP có hình dạng tùy ý và lý tưởng sẽ nhận được một vector chuyển động duy nhất nhất quán với chuyển động của đối tượng.

- Mã hóa dựa trên VOP của MPEG-4 cũng sử dụng kỹ thuật bù chuyển động.

- Một VOP được mã hóa intraframe được gọi là I-VOP. Các VOP được mã hóa interframe được gọi là P-VOP nếu chỉ sử dụng dự đoán tiến và B-VOP nếu sử dụng dự đoán hai chiều. Khó khăn mới ở đây là các VOP có thể có hình dạng tùy ý. Do đó, ngoài thông tin về kết cấu, thông tin hình dạng cũng cần được mã hóa. Cần lưu ý rằng kết cấu ở đây thực sự đề cập đến nội dung hình ảnh, tức là giá trị độ xám và giá trị màu sắc của các pixel trong VOP. MPEG-1 và 2 không mã hóa thông tin hình dạng, vì tất cả các khung hình đều hình chữ nhật, nhưng họ mã hóa giá trị của các pixel trong khung hình. Trong MPEG-1 và 2, việc mã hóa này không được gọi một cách rõ ràng là mã hóa kết cấu. Thuật ngữ "kết cấu" xuất phát từ đồ họa máy tính và cho thấy cách mà lĩnh vực này đã vào thế giới mã hóa video với MPEG-4.

Dưới đây, chúng ta sẽ bắt đầu với một cuộc thảo luận về mã hóa dựa trên bù chuyển động cho các VOP, tiếp theo là giới thiệu về mã hóa kết cấu, mã hóa hình dạng, mã hóa kết cấu tĩnh, mã hóa sprite và bù chuyển động toàn cầu.

**Bù chuyển động ( Motion Compensation )**

- Phần này giải quyết các vấn đề liên quan đến bù chuyển động dựa trên VOP trong MPEG-4. Do việc mã hóa I-VOP tương đối đơn giản, các thảo luận của chúng tôi sẽ tập trung vào việc mã hóa cho P-VOP và/hoặc B-VOP trừ khi I-VOP được đề cập cụ thể.

- Như trước đây, mã hóa VOP dựa trên bù chuyển động trong MPEG-4 lại bao gồm ba bước: ước lượng chuyển động, dự đoán dựa trên bù chuyển động, và mã hóa lỗi dự đoán. Để tạo điều kiện cho việc bù chuyển động, mỗi VOP được chia thành nhiều macroblock, giống như trong các phương pháp dựa trên khung hình trước đó. Các macroblock mặc định có kích thước 16×16 trong hình ảnh độ sáng và 8×8 trong hình ảnh độ màu, và được xử lý đặc biệt khi chúng vượt qua ranh giới của một VOP có hình dạng tùy ý.

A diagram of a vehicle

Description automatically generated

- MPEG-4 định nghĩa một hộp giới hạn hình chữ nhật cho mỗi VOP. Các ranh giới bên trái và bên trên của nó là các ranh giới bên trái và bên trên của VOP, qua đó xác định điểm gốc đã dịch chuyển cho VOP từ (0, 0) ban đầu cho khung hình video trong hệ tọa độ tuyệt đối (khung hình) (xem Hình 11.15). Cả chiều ngang và chiều dọc của hộp giới hạn phải là bội số của 16 trong hình ảnh độ sáng. Do đó, hộp này thường lớn hơn một chút so với một hộp giới hạn thông thường.

A diagram of a paddling box

Description automatically generated

- Các macroblock hoàn toàn nằm trong VOP được gọi là macroblock bên trong. Như có thể thấy từ Hình 11.15, nhiều macroblock nằm trên ranh giới của VOP và được gọi là macroblock biên.

- Việc bù đắp chuyển động cho các macroblock bên trong được thực hiện theo cách tương tự như trong MPEG-1 và 2. Tuy nhiên, các macroblock biên có thể khó khớp trong ước lượng chuyển động, vì các VOP thường có hình dạng tùy ý (không hình chữ nhật), và hình dạng của chúng có thể thay đổi từ khoảnh khắc này sang khoảnh khắc khác trong video. Để giúp khớp mọi pixel trong VOP mục tiêu và đáp ứng yêu cầu bắt buộc về các khối hình chữ nhật trong mã hóa biến đổi (ví dụ: DCT), một bước tiền xử lý là lấp đầy được áp dụng cho các VOP tham chiếu trước khi ước lượng chuyển động.

- Chỉ có các pixel nằm trong VOP của VOP hiện tại (mục tiêu) được xem xét để khớp trong việc bù đắp chuyển động, và việc lấp đầy chỉ diễn ra ở các VOP tham chiếu.

- Về chất lượng, có thể đã phát triển một phương pháp ngoại suy tốt hơn so với việc lấp đầy. Tuy nhiên, việc lấp đầy đã được áp dụng trong MPEG-4 chủ yếu do tính đơn giản và tốc độ của nó.

- Hai bước đầu tiên của việc bù đắp chuyển động là: lấp đầy và mã hóa vector chuyển động.

**- Lấp đầy (Padding).** Đối với tất cả các macroblock biên trong VOP tham chiếu, lấp đầy lặp lại theo chiều ngang được thực hiện trước, sau đó là lấp đầy lặp lại theo chiều dọc (Hình 11.16).

- Sau đó, đối với tất cả các macroblock bên ngoài nằm ngoài VOP nhưng liền kề với một hoặc nhiều macroblock biên, lấp đầy mở rộng được áp dụng.

- Thuật toán lấp đầy lặp lại theo chiều ngang xem xét từng hàng trong các macroblock biên trong VOP tham chiếu. Mỗi pixel biên được sao chép sang bên trái và/hoặc bên phải để điền các giá trị cho khoảng pixel nằm ngoài VOP trong macroblock. Nếu khoảng này được giới hạn bởi hai pixel biên, thì giá trị trung bình của chúng sẽ được áp dụng.

A grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with a grid with

Description automatically generated

A black and white text on a white background

Description automatically generated

- Thuật toán lấp đầy lặp lại theo chiều dọc tiếp theo hoạt động tương tự. Nó xem xét từng cột, và các pixel mới được lấp đầy bởi quá trình lấp đầy ngang trước đó được coi là pixel bên trong VOP cho mục đích lấp đầy dọc này.

**Example 11.1**

- Hình 11.17 minh họa một ví dụ về việc lấp đầy lặp lại trong một macroblock biên của một VOP tham chiếu. Hình 11.17a cho thấy các giá trị cường độ (hoặc sắc độ) của các pixel trong VOP, với biên của VOP được thể hiện bằng các đường tối màu. Để đơn giản, độ phân giải của macroblock đã được giảm xuống còn 6 × 6 trong ví dụ này, mặc dù kích thước thực tế của macroblock là 16 × 16 trong hình ảnh độ sáng và 8 × 8 trong hình ảnh sắc độ.

1. **Lấp đầy lặp lại theo chiều ngang (Hình 11.17b)**

* **Hàng 0**: Pixel bên phải nhất của VOP là pixel biên duy nhất. Giá trị cường độ của nó, 60, được sử dụng lặp lại như giá trị của các pixel bên ngoài VOP.
* **Hàng 1**: Tương tự, pixel bên phải nhất của VOP là pixel biên duy nhất. Giá trị cường độ của nó, 50, được sử dụng lặp lại như giá trị pixel bên ngoài VOP.
* **Hàng 2 và 3**: Không có lấp đầy theo chiều ngang, vì không có pixel biên nào.
* **Hàng 4**: Có hai khoảng bên ngoài VOP, mỗi khoảng được giới hạn bởi một pixel biên duy nhất. Giá trị cường độ của chúng, 60 và 70, được sử dụng làm giá trị pixel cho hai khoảng này, tương ứng.
* **Hàng 5**: Một khoảng duy nhất bên ngoài VOP được giới hạn bởi một cặp pixel biên của VOP. Giá trị trung bình của các giá trị cường độ của chúng, (50 + 80)/2 = 65, được sử dụng lặp lại như giá trị của các pixel ở giữa chúng.

1. **Lấp đầy lặp lại theo chiều dọc (Hình 11.17c)**

* **Cột 0**: Một khoảng duy nhất được giới hạn bởi một cặp pixel biên của VOP. Một pixel có giá trị 42 trong VOP; pixel còn lại là 60, vừa được tạo ra từ việc lấp đầy theo chiều ngang. Giá trị trung bình của các giá trị cường độ của chúng, (42 + 60)/2 = 51, được sử dụng lặp lại làm giá trị của các pixel ở giữa chúng.
* **Các cột 1, 2, 3, 4 và 5**: Các cột này được lấp đầy tương tự như Cột 0.

**- Lấp đầy mở rộng.** Các macroblock hoàn toàn nằm ngoài VOP được gọi là macroblock ngoại. Các macroblock ngoại ngay bên cạnh các macroblock biên được lấp đầy bằng cách sao chép giá trị của các pixel biên của macroblock biên. Chúng ta lưu ý rằng các macroblock biên hiện đã được lấp đầy hoàn toàn, vì vậy tất cả các pixel biên theo chiều ngang và chiều dọc của chúng đều có giá trị xác định. Nếu một macroblock ngoại có nhiều hơn một macroblock biên làm hàng xóm liền kề, macroblock biên được sử dụng cho lấp đầy mở rộng sẽ theo thứ tự ưu tiên: trái, trên, phải và dưới.

- Các phiên bản sau của MPEG-4 cho phép một số giá trị trung bình của các macroblock này được sử dụng. Quá trình lấp đầy mở rộng này có thể được lặp lại để lấp đầy tất cả các macroblock ngoại trong hộp bao hình chữ nhật của VOP.

**- Mã hóa vector chuyển động.** Mỗi macroblock từ VOP mục tiêu sẽ tìm một macroblock phù hợp nhất từ VOP tham chiếu thông qua quy trình ước lượng chuyển động sau đây:

- Gọi C(x+k,y+l)C(x + k, y + l)C(x+k,y+l) là các pixel của macroblock trong VOP mục tiêu, và R(x+i+k,y+j+l)R(x + i + k, y + j + l)R(x+i+k,y+j+l) là các pixel của macroblock trong VOP tham chiếu. Tương tự như MAD trong Eq. (10.1), một Tổng độ chênh lệch tuyệt đối (SAD) để đo lường sự khác biệt giữa hai macroblock có thể được định nghĩa như sau:

A math equation with black text

Description automatically generated with medium confidence

Trong đó N là kích thước của macroblock. Map (p,q) = 1 khi C(p,q) là một pixel nằm trong VOP mục tiêu; ngược lại, Map(p,q) = 0. Vecto (i,j) tạo ra giá trị SAD tối thiểu sẽ được chọn làm vector chuyển động MV(u,v):



Trong đó p là độ lớn tối đa cho phép của u và v.

- Để bù đắp chuyển động, vector chuyển động MV được mã hóa. Giống như trong H.263, vector chuyển động của macroblock mục tiêu không đơn thuần được lấy làm MV. Thay vào đó, MV được dự đoán từ ba macroblock lân cận. Lỗi dự đoán cho vector chuyển động sau đó được mã hóa theo chiều dài biến.

- Sau đây là một số kỹ thuật bù đắp chuyển động tiên tiến được áp dụng tương tự như trong H.263:

* Bốn vector chuyển động (mỗi vector từ một khối 8 × 8) có thể được tạo ra cho mỗi macroblock trong thành phần độ sáng của một VOP.
* Vector chuyển động có thể có độ chính xác dưới pixel. Ở độ chính xác nửa pixel, phạm vi của vector chuyển động là [−2,048, 2,047]. MPEG-4 cũng cho phép độ chính xác pixel tư quý trong thành phần độ sáng của một VOP.
* Vector chuyển động không bị giới hạn được cho phép: MV có thể chỉ ra vượt ra ngoài ranh giới của VOP tham chiếu. Khi một pixel ngoài VOP được tham chiếu, giá trị của nó vẫn được định nghĩa nhờ vào việc lấp đầy.

**Mã hóa kết cấu.**

- Kết cấu đề cập đến sự biến đổi mức xám (hoặc màu) và/hoặc các mẫu trong VOP. Mã hóa kết cấu trong MPEG-4 có thể dựa trên DCT hoặc DCT thích ứng với hình dạng (SA-DCT).

**- Mã hóa kết cấu dựa trên DCT.** Trong I-VOP, các giá trị xám (hoặc màu) của các pixel trong mỗi macroblock của VOP được mã hóa trực tiếp, sử dụng DCT theo sau là VLC, điều này tương tự như những gì được thực hiện trong JPEG cho hình ảnh tĩnh. P-VOP và B-VOP sử dụng mã hóa dựa trên bù trừ chuyển động; do đó, là lỗi dự đoán được gửi đến DCT và VLC. Cuộc thảo luận tiếp theo sẽ tập trung vào mã hóa kết cấu dựa trên bù trừ chuyển động cho P-VOP và B-VOP.

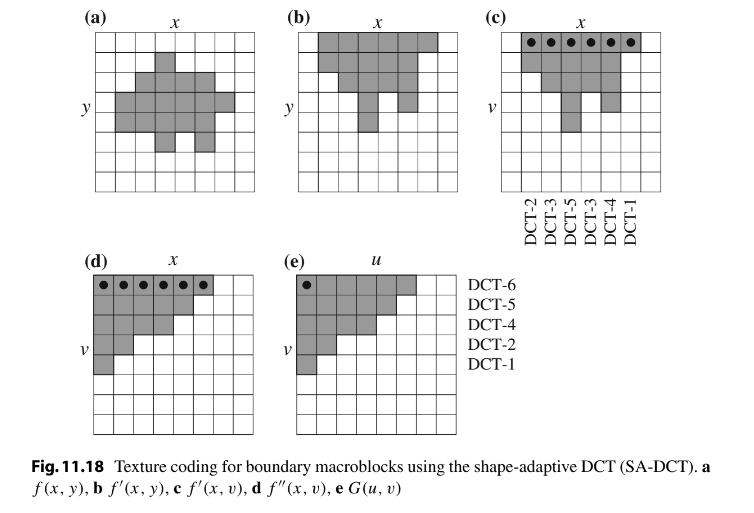
- Mã hóa cho các macroblock nội bộ, mỗi macroblock có kích thước 16 × 16 trong VOP độ sáng và 8 × 8 trong VOP độ sắc, tương tự như mã hóa dựa trên bù trừ chuyển động thông thường trong H.261, H.263, và MPEG-1 và 2. Các lỗi dự đoán từ sáu khối 8 × 8 của mỗi macroblock được thu thập sau bước ước lượng chuyển động thông thường. Những lỗi này được gửi đến một quy trình DCT để thu được sáu khối hệ số DCT 8 × 8.

- Đối với các macroblock biên, các khu vực bên ngoài VOP trong VOP tham chiếu được đệm bằng phương pháp đệm lặp lại, như đã mô tả ở trên. Sau khi bù trừ chuyển động, các lỗi dự đoán kết cấu trong VOP mục tiêu được thu thập. Đối với các phần của macroblock biên trong VOP mục tiêu nằm ngoài VOP, các giá trị bằng không được thêm vào khối gửi đến DCT, vì lý tưởng, các lỗi dự đoán sẽ gần bằng không bên trong VOP. Trong khi đệm lặp lại và đệm mở rộng được sử dụng để cải thiện việc khớp trong bù trừ chuyển động, việc thêm đệm bằng không này nhằm cải thiện kết quả DCT trong mã hóa kết cấu.

- Bước lượng hóa step\_size cho thành phần DC là 8. Đối với các hệ số AC, có thể sử dụng một trong hai phương pháp sau:

* Phương pháp H.263, trong đó tất cả các hệ số nhận cùng một bộ điều chỉnh lượng hóa được kiểm soát bởi một tham số duy nhất, và các macroblock khác nhau có thể có các bộ điều chỉnh khác nhau.
* Phương pháp MPEG-2, trong đó các hệ số DCT trong cùng một macroblock có thể có các bộ điều chỉnh khác nhau và được kiểm soát thêm bởi tham số step\_size.

**- Mã hóa dựa trên DCT thích ứng với hình dạng (SA-DCT)** là một phương pháp mã hóa kết cấu khác cho các macroblock biên. Do hiệu quả của nó, SA-DCT đã được áp dụng cho việc mã hóa các macroblock biên trong phiên bản 2 của MPEG-4.



- 1D DCT-N là một biến thể của DCT 1 chiều được mô tả trước đó [Eqs. (8.19) và (8.20)], trong đó N phần tử được sử dụng trong phép biến đổi thay vì một N cố định là 8 (Để rút gọn, chúng tôi sẽ ký hiệu phép biến đổi 1D DCT-N bằng DCT-N trong phần này).

- Các phương trình (11.4) và (11.5) mô tả phép biến đổi DCT-N và phép biến đổi ngược của nó, IDCT-N.

**Biến đổi Cosin rời rạc 1 chiều-N (DCT-N)**

**A math equations and numbers

Description automatically generated with medium confidence**

**Biến đổi Cosin rời rạc ngược 1 chiều-N (IDCT-N)**

**A math equations and formulas

Description automatically generated**

- SA-DCT là một biến đổi 2D và được tính toán như một biến đổi 2D tách rời qua hai lần lặp DCT-N. Hình 11.18 minh họa quá trình mã hóa kết cấu cho các macroblock biên sử dụng SA-DCT. Biến đổi được áp dụng cho từng khối 8 × 8 trong macroblock biên.

- Hình 11.18a cho thấy một trong các khối 8 × 8 của một macroblock biên, nơi các pixel bên trong macroblock, ký hiệu là f(x, y), được hiển thị bằng màu xám. Các pixel xám đầu tiên được dịch lên trên để thu được f≤(x, y), như hình 11.18b cho thấy. Trong lần lặp đầu tiên, DCT-N được áp dụng cho từng cột của f≤(x, y), với N được xác định bởi số lượng pixel xám trong cột. Do đó, chúng ta sử dụng DCT-2, DCT-3, DCT-5, và như vậy. Các hệ số DCT-N thu được được ký hiệu là F≤(x, v), như hình 11.18c cho thấy, trong đó các chấm tối biểu thị các hệ số DC của DCT-N. Các phần tử của F≤(x, v) sau đó được dịch sang trái để thu được F≤≤(x, v) trong hình 11.18d.

- Trong lần lặp thứ hai, DCT-N được áp dụng cho từng hàng của F≤≤(x, v) để thu được G(u, v) (hình 11.18e), trong đó chấm tối đơn lẻ biểu thị hệ số DC G(0, 0) của 2D SA-DCT.

Một số cân nhắc về mã hóa:

* Tổng số hệ số DCT trong G(u, v) bằng với số pixel xám bên trong khối 8×8 của macroblock biên, nhỏ hơn 8×8. Do đó, phương pháp này là thích ứng với hình dạng và hiệu quả hơn trong tính toán.
* Tại thời điểm giải mã, vì các phần tử của mảng phải được dịch trở lại đúng cách sau mỗi lần lặp của IDCT-N, một mặt nạ nhị phân có hình dạng ban đầu là cần thiết để giải mã thông tin kết cấu được mã hóa bằng SA-DCT. Mặt nạ nhị phân giống như bản đồ alpha nhị phân được mô tả bên dưới.

**Mã hóa hình dạng**

- Khác với MPEG-1 và MPEG-2, MPEG-4 phải mã hóa hình dạng của VOP, vì hình dạng là một trong những đặc điểm nội tại của các đối tượng hình ảnh.

- MPEG-4 hỗ trợ hai loại thông tin hình dạng: nhị phân và độ xám. Thông tin hình dạng nhị phân có thể dưới dạng một bản đồ nhị phân (còn được gọi là bản đồ alpha nhị phân) có kích thước bằng với bounding box hình chữ nhật của VOP. Giá trị 1 (mờ) hoặc 0 (trong suốt) trong bitmap cho biết pixel nằm trong hoặc ngoài VOP. Ngoài ra, thông tin hình dạng độ xám thực sự đề cập đến độ trong suốt của hình dạng, với các giá trị xám dao động từ 0 (trong suốt) đến 255 (mờ).

**- Mã hóa hình dạng nhị phân**. Để mã hóa bản đồ alpha nhị phân một cách hiệu quả hơn, bản đồ được chia thành các khối 16 × 16, còn được gọi là Các Khối Alpha Nhị Phân (BAB). Nếu một BAB hoàn toàn mờ hoặc trong suốt, thì việc mã hóa trở nên dễ dàng và không cần kỹ thuật mã hóa hình dạng đặc biệt nào. Các BAB biên chứa đường viền và do đó thông tin hình dạng cho VOP. Chúng là đối tượng của mã hóa hình dạng nhị phân.

- Nhiều thuật toán dựa trên đường viền và thuật toán dựa trên bitmap (hoặc dựa trên diện tích) đã được nghiên cứu và so sánh để mã hóa các BAB biên. Hai trong số các ứng viên cuối cùng đều là các thuật toán dựa trên bitmap. Một là thuật toán READ đã được sửa đổi (MMR), cũng là một nâng cấp tùy chọn trong tiêu chuẩn fax Nhóm 3 (G3) và là phương pháp nén bắt buộc trong tiêu chuẩn Nhóm 4 (G4). Ứng viên còn lại là Mã hóa số học dựa trên ngữ cảnh (CAE), được phát triển ban đầu cho JBIG. Cuối cùng, CAE được chọn làm phương pháp mã hóa hình dạng nhị phân cho MPEG-4 do sự đơn giản và hiệu quả nén của nó.

A screenshot of a test

Description automatically generated

- MMR về cơ bản là một chuỗi đơn giản hóa của thuật toán Relative Element Address Designate (READ). Ý tưởng cơ bản phía sau thuật toán READ là mã hóa dòng hiện tại tương đối với các vị trí pixel trong dòng đã được mã hóa trước đó. Thuật toán bắt đầu bằng cách xác định năm vị trí pixel trong dòng trước và dòng hiện tại:

* **a0**: giá trị pixel cuối cùng mà encoder và decoder đều biết
* **a1**: pixel chuyển tiếp bên phải a0
* **a2**: pixel chuyển tiếp thứ hai bên phải a0
* **b1**: pixel chuyển tiếp đầu tiên có màu đối lập với a0 trong dòng đã được mã hóa trước đó
* **b2**: pixel chuyển tiếp đầu tiên bên phải b1 trong dòng đã được mã hóa trước đó

- READ hoạt động bằng cách kiểm tra các vị trí tương đối của các pixel này. Tại bất kỳ thời điểm nào, cả encoder và decoder đều biết vị trí của a0, b1 và b2, trong khi các vị trí a1 và a2 chỉ được biết trong encoder.

- Ba chế độ mã hóa được sử dụng:

* Nếu độ dài chạy trong dòng trước và dòng hiện tại tương tự nhau, khoảng cách giữa a1 và b1 sẽ nhỏ hơn nhiều so với khoảng cách giữa a0 và a1. Do đó, chế độ dọc mã hóa độ dài chạy hiện tại dưới dạng a1 - b1.
* Nếu dòng trước không có độ dài chạy tương tự, độ dài chạy hiện tại sẽ được mã hóa bằng mã hóa độ dài chạy một chiều. Đây được gọi là chế độ ngang.
* Nếu **a0 ⇒ b1 < b2 < a1**, chúng ta có thể đơn giản truyền một mã số chỉ ra rằng nó đang ở chế độ qua và tiến a0 đến vị trí dưới b2, sau đó tiếp tục quá trình mã hóa.

- Một số đơn giản hóa có thể được thực hiện đối với thuật toán READ để triển khai thực tế. Ví dụ, nếu ∣a1−b1∣<3|a1 - b1| < 3∣a1−b1∣<3, thì chỉ cần chỉ ra rằng chúng ta có thể áp dụng chế độ dọc. Ngoài ra, để ngăn ngừa sự truyền lỗi, một yếu tố k được định nghĩa, sao cho mỗi k dòng phải chứa ít nhất một dòng được mã hóa bằng mã hóa độ dài chạy thông thường. Những sửa đổi này tạo thành thuật toán Modified READ được sử dụng trong tiêu chuẩn G3. Thuật toán Modified Modified READ (MMR) đơn giản loại bỏ các hạn chế do yếu tố k đặt ra.

- Đối với Mã hóa số học dựa trên ngữ cảnh (Context-based Arithmetic Encoding), Hình 11.19 minh họa "ngữ cảnh" cho một pixel trong Binary Alpha Block (BAB) biên. Trong chế độ intra-CAE, khi chỉ có alpha map mục tiêu liên quan (Hình 11.19a), mười pixel lân cận (được đánh số từ 0 đến 9) trong cùng một alpha map tạo thành ngữ cảnh. Mười số nhị phân liên quan đến những pixel này có thể cung cấp tới 2^10 = 1,024 ngữ cảnh khả thi.

- Bây giờ, rõ ràng rằng một số ngữ cảnh nhất định (ví dụ: tất cả là 1 hoặc tất cả là 0) xuất hiện thường xuyên hơn những ngữ cảnh khác. Với một số thống kê trước, một bảng xác suất có thể được xây dựng để chỉ ra xác suất xảy ra cho mỗi ngữ cảnh trong số 1.024 ngữ cảnh.

- Nhớ rằng Mã hóa số học (Chương 7) có khả năng mã hóa một chuỗi các ký hiệu xác suất bằng một số duy nhất. Bây giờ, mỗi pixel có thể tra cứu bảng để tìm giá trị xác suất cho ngữ cảnh của nó. CAE đơn giản quét từng pixel 16 × 16 trong mỗi Binary Alpha Block (BAB) theo thứ tự và áp dụng mã hóa số học để cuối cùng tạo ra một số điểm động duy nhất cho BAB.

- Chế độ Inter-CAE là một mở rộng tự nhiên của intra-CAE: nó liên quan đến cả alpha map mục tiêu và alpha map tham chiếu. Đối với mỗi macroblock biên trong khung mục tiêu, một quy trình ước lượng chuyển động (với độ chính xác nguyên) và bù chuyển động được thực hiện trước để xác định macroblock phù hợp trong khung tham chiếu. Điều này thiết lập các vị trí tương ứng cho từng pixel trong BAB biên.

- Hình 11.19b cho thấy ngữ cảnh của mỗi pixel bao gồm bốn pixel lân cận từ alpha map mục tiêu và năm pixel từ alpha map tham chiếu. Dựa trên ngữ cảnh của nó, mỗi pixel trong Binary Alpha Block (BAB) được gán một trong 2^9 = 512 xác suất. Sau đó, thuật toán CAE được áp dụng.

- Bản đồ nhị phân 16×16 ban đầu chứa 256 bit thông tin. Việc nén nó thành một số duy nhất đạt được một sự tiết kiệm đáng kể.

- Phương pháp CAE nêu trên là không mất dữ liệu! Nhóm MPEG-4 cũng đã xem xét một số phiên bản đơn giản mất dữ liệu của phương pháp mã hóa hình dạng nêu trên. Ví dụ, alpha map nhị phân có thể đơn giản được giảm mẫu với hệ số 2 hoặc 4 trước khi mã hóa số học. Tất nhiên, sự đánh đổi là sự suy giảm của hình dạng.

- **Mã hóa hình dạng mức xám.** Thuật ngữ mã hóa hình dạng mức xám trong MPEG-4 có thể gây nhầm lẫn, vì thông tin hình dạng thực sự được mã hóa trong alpha map nhị phân. Mức xám ở đây được sử dụng để mô tả độ trong suốt của hình dạng, không phải kết cấu!

- Ngoài các bitplane cho bộ đệm khung RGB, đồ họa raster sử dụng thêm các bitplane cho alpha map, có thể được sử dụng để mô tả độ trong suốt của đối tượng đồ họa. Khi alpha map có nhiều hơn một bitplane, nhiều cấp độ trong suốt có thể được giới thiệu—ví dụ, 0 cho trong suốt, 255 cho không trong suốt, và bất kỳ số nào ở giữa cho các mức độ trong suốt trung gian khác nhau.

- Thuật ngữ mức xám được sử dụng cho mã hóa độ trong suốt trong MPEG-4 đơn giản là vì số độ trong suốt tình cờ nằm trong khoảng từ 0 đến 255—giống như cường độ mức xám 8-bit thông thường.

- Mã hóa hình dạng mức xám trong MPEG-4 sử dụng cùng một kỹ thuật như trong mã hóa kết cấu đã được mô tả ở trên. Nó sử dụng alpha map và bù trừ chuyển động dựa trên khối và mã hóa các lỗi dự đoán bằng DCT. Các macroblock biên cần phải được lấp đầy, như trước đây, vì không phải tất cả các pixel đều nằm trong VOP.

- Việc mã hóa thông tin độ trong suốt (mã hóa hình dạng mức xám) là mất dữ liệu, trái ngược với việc mã hóa thông tin hình dạng nhị phân, vốn mặc định là không mất dữ liệu.

**Mã hóa kết cấu tĩnh**

- MPEG-4 sử dụng mã hóa wavelet cho kết cấu của các đối tượng tĩnh, đặc biệt phù hợp khi kết cấu được dùng để ánh xạ lên các bề mặt 3D.

- Như đã giới thiệu ở Chương 8, mã hóa wavelet có thể phân rã hình ảnh thành các dải con của nhiều tần số khác nhau một cách đệ quy. Thuật toán Embedded Zerotree Wavelet (EZW) cung cấp một cách biểu diễn gọn gàng bằng cách tận dụng số lượng lớn các hệ số không đáng kể trong các dải con.

- Mã hóa các dải con trong mã hóa kết cấu tĩnh MPEG-4 được thực hiện như sau:

* Các dải tần số thấp nhất được mã hóa bằng DPCM, trong đó mỗi hệ số được dự đoán dựa trên ba hệ số lân cận.
* Các dải tần số khác được mã hóa dựa trên phương pháp mã hóa wavelet cây không-đỉnh đa tỷ lệ.

- Cây không-đỉnh đa tỷ lệ có mối quan hệ cha–con (PCR) cho mỗi hệ số trong dải tần số thấp nhất. Kết quả là thông tin vị trí của tất cả các hệ số được theo dõi tốt hơn.

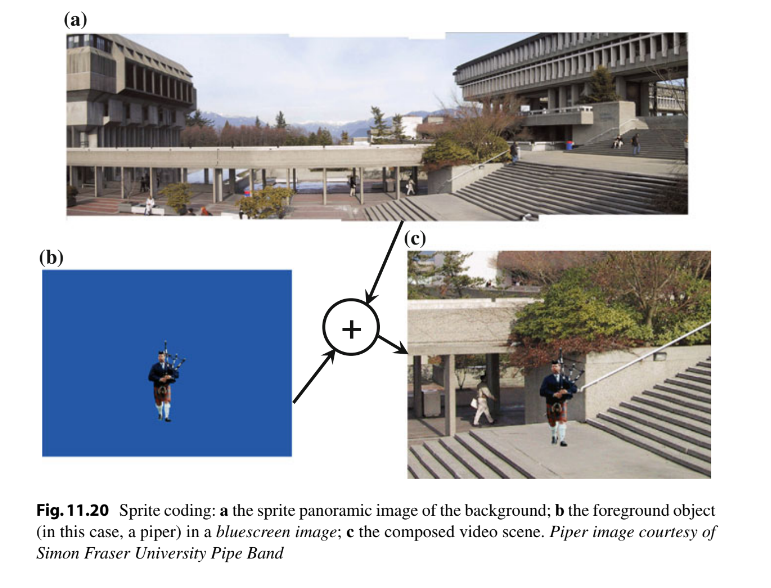
- Ngoài độ lớn ban đầu của các hệ số, mức độ lượng tử hóa cũng ảnh hưởng đến tốc độ dữ liệu. Nếu độ lớn của một hệ số bằng không sau khi lượng tử hóa, nó được coi là không đáng kể. Ban đầu, một bộ lượng tử lớn được sử dụng; chỉ các hệ số có ý nghĩa nhất được chọn và mã hóa bằng mã hóa số học. Phần chênh lệch giữa hệ số lượng tử hóa và hệ số ban đầu được giữ trong các dải phụ dư, sẽ được mã hóa trong lần lặp tiếp theo với một bộ lượng tử nhỏ hơn. Quá trình này có thể tiếp tục trong các lần lặp bổ sung, do đó rất linh hoạt.

**Mã hóa Sprite**

- Quay video thường bao gồm các chuyển động của máy ảnh như lia ngang, nghiêng, phóng to/thu nhỏ, v.v. Thông thường, mục tiêu chính là theo dõi và kiểm tra các đối tượng (di chuyển) ở tiền cảnh. Trong những trường hợp này, phần nền có thể được coi là một hình ảnh tĩnh. Điều này tạo ra một loại Đối tượng Hình ảnh (VO) mới, gọi là sprite—một hình ảnh đồ họa có thể di chuyển tự do trong một hình ảnh đồ họa lớn hơn hoặc tập hợp các hình ảnh.

- Để tách đối tượng tiền cảnh khỏi nền, chúng ta giới thiệu khái niệm sprite panorama—một hình ảnh tĩnh mô tả nền tĩnh qua một chuỗi khung hình video. Nó có thể được tạo ra bằng cách sử dụng kỹ thuật “stitching” và warping [20]. Hình ảnh sprite panoramic lớn có thể được mã hóa và gửi đến bộ giải mã chỉ một lần, ở phần đầu của chuỗi video. Khi bộ giải mã nhận các đối tượng tiền cảnh được mã hóa riêng và các tham số mô tả chuyển động của máy ảnh đến thời điểm đó, nó có thể tái tạo cảnh một cách hiệu quả.

- Hình 11.20a cho thấy một sprite là một hình ảnh panoramic được ghép từ một chuỗi các khung hình video. Bằng cách kết hợp nền sprite với nhân vật trong hình ảnh bluescreen (Hình 11.20b), cảnh video mới (Hình 11.20c) có thể dễ dàng được giải mã với sự trợ giúp của mã sprite và các tham số pan/tilt cùng zoom bổ sung. Rõ ràng, các đối tượng tiền cảnh có thể là từ cảnh video gốc hoặc được tạo ra mới để thực hiện việc kết hợp đối tượng linh hoạt trong video MPEG-4.



**Bù đắp chuyển động toàn cục**

- Các chuyển động máy quay phổ biến, như quay ngang (pan), nghiêng (tilt), xoay và phóng to (zoom) (còn gọi là chuyển động toàn cục, vì chúng áp dụng cho tất cả các khối), thường gây ra sự thay đổi nội dung nhanh chóng giữa các khung hình video kế tiếp. Bù đắp chuyển động theo khối truyền thống sẽ dẫn đến một số lượng lớn các vector chuyển động có ý nghĩa. Hơn nữa, những loại chuyển động máy quay này không thể hoàn toàn được mô tả bằng mô hình chuyển động dịch chuyển (translational motion model) mà bù đắp chuyển động theo khối sử dụng. Bù đắp chuyển động toàn cục (GMC) được thiết kế để giải quyết vấn đề này. Có bốn thành phần chính:

• **Ước lượng chuyển động toàn cục**. Ước lượng chuyển động toàn cục tính toán chuyển động của hình ảnh hiện tại so với sprite. "Toàn cục" ở đây có nghĩa là sự thay đổi tổng thể do thay đổi của máy quay—như phóng to, quay ngang, và các chuyển động tương tự. Nó được tính bằng cách tối thiểu hóa tổng các sai số bình phương giữa sprite S và hình ảnh đã bù đắp chuyển động toàn cục I’.

A black text on a white background

Description automatically generated

- Ý tưởng ở đây là nếu hình ảnh nền (có thể là ảnh ghép) là một sprite S(xi, yi), chúng ta mong đợi khung hình mới chủ yếu gồm nền giống như vậy, được thay đổi bởi các chuyển động toàn cục của máy quay. Để ràng buộc thêm cho bài toán ước lượng chuyển động toàn cục, chuyển động trên toàn bộ hình ảnh được tham số hóa bởi một mô hình chuyển động phối cảnh sử dụng tám tham số, được định nghĩa như sau:

A group of math symbols

Description automatically generated

- Vấn đề tối ưu hóa có ràng buộc này có thể được giải quyết bằng phương pháp dựa trên gradient-descent [21].

* **Biến dạng và pha trộn**: Sau khi các tham số chuyển động được tính toán, các hình ảnh nền sẽ được biến dạng để phù hợp với sprite. Tọa độ của hình ảnh biến dạng được tính toán bằng công thức (11.7). Sau đó, hình ảnh biến dạng sẽ được pha trộn vào sprite hiện tại để tạo ra sprite mới. Quá trình này có thể được thực hiện bằng cách trung bình đơn giản hoặc một hình thức trung bình có trọng số.
* **Mã hóa quỹ đạo chuyển động**: Thay vì truyền trực tiếp các tham số chuyển động, ta chỉ mã hóa các sự dịch chuyển của các điểm tham chiếu. Đây được gọi là mã hóa quỹ đạo chuyển động [21]. Các điểm ở các góc của hộp chứa VOP được sử dụng làm điểm tham chiếu, và các điểm tương ứng của chúng trong sprite sẽ được tính toán. Sự khác biệt giữa hai thực thể này được mã hóa và truyền đi dưới dạng các vector chuyển động vi phân.
* **Lựa chọn mã hóa chuyển động cục bộ (LMC) hay mã hóa chuyển động toàn cục (GMC)**: Cuối cùng, cần phải đưa ra quyết định sử dụng GMC hay LMC. Để làm điều này, ta có thể áp dụng GMC cho nền chuyển động và LMC cho đối tượng tiền cảnh. Theo cách đánh giá một cách ngắn gọn (và bỏ qua nhiều chi tiết), nếu **SAD(GMC) < SAD(LMC)**, thì sử dụng GMC để tạo ra VOP tham chiếu dự đoán. Ngược lại, sử dụng LMC như trước đây.

**11.4.3. Mã hóa đối tượng tổng hợp trong MPEG-4:**

- Số lượng đối tượng trong video được tạo ra bằng đồ họa máy tính và phần mềm hoạt hình đang ngày càng tăng. Những đối tượng này được gọi là đối tượng tổng hợp và thường có thể được trình bày cùng với các đối tượng và cảnh tự nhiên trong các trò chơi, quảng cáo TV, chương trình, và phim hoạt hình hoặc phim truyện.  
- Trong phần này, chúng ta sẽ thảo luận ngắn gọn về các phương pháp mã hóa và hoạt hình đối tượng tổng hợp dựa trên lưới 2D và mô hình 3D. Beek, Petajan, và Ostermann [22] cung cấp một khảo sát chi tiết hơn về chủ đề này.

**Mã hóa đối tượng lưới 2D**

- Lưới 2D là một sự phân vùng (hoặc phân tách) của một vùng phẳng 2D bằng các mảnh đa giác. Các đỉnh của các đa giác này được gọi là các nút của lưới. Các lưới phổ biến nhất là lưới tam giác, trong đó tất cả các đa giác đều là tam giác. Tiêu chuẩn MPEG-4 sử dụng hai loại lưới 2D: lưới đồng nhất và lưới Delaunay [23]. Cả hai đều là lưới tam giác và có thể được sử dụng để mô hình hóa các đối tượng video tự nhiên cũng như các đối tượng hoạt hình tổng hợp.

A diagram of a mesh object

Description automatically generated

- Vì cấu trúc phân hoạch tam giác (các cạnh giữa các nút) đã được biết đến và có thể dễ dàng tái tạo bởi bộ giải mã, nó không được mã hóa một cách rõ ràng trong dòng bit. Do đó, mã hóa đối tượng lưới 2D rất gọn nhẹ. Tất cả các giá trị tọa độ của lưới được mã hóa với độ chính xác nửa pixel.

- Mỗi lưới 2D được xem như một mặt phẳng đối tượng lưới (MOP). Hình 11.21 minh họa quá trình mã hóa cho MOP 2D. Quá trình mã hóa có thể được chia thành mã hóa hình học và mã hóa chuyển động. Như đã chỉ ra, dữ liệu đầu vào là tọa độ x và y của tất cả các nút và các tam giác (tm) trong lưới. Dữ liệu đầu ra là các sự thay đổi (dxn, dyn) và lỗi dự đoán chuyển động (exn, eyn), cả hai đều được giải thích dưới đây.

- **Mã hóa hình học lưới 2D.** MPEG-4 cho phép bốn loại lưới đồng đều với các cấu trúc tam giác khác nhau. Hình 11.22 hiển thị các lưới như vậy với 4 × 5 nút lưới. Mỗi lưới đồng đều có thể được chỉ định bởi năm tham số: hai tham số đầu tiên chỉ số lượng nút trong mỗi hàng và cột, tương ứng; hai tham số tiếp theo chỉ kích thước ngang và dọc của mỗi hình chữ nhật (chứa hai tam giác), tương ứng; và tham số cuối cùng chỉ loại lưới đồng đều.

- Lưới đồng đều đơn giản và đặc biệt tốt trong việc đại diện cho các đối tượng hình chữ nhật 2D (ví dụ: toàn bộ khung video). Khi sử dụng cho các đối tượng có hình dạng tùy ý, chúng được áp dụng (chồng lên) các hộp bao quanh của các VOP, điều này gây ra một số sự không hiệu quả.

- Lưới Delaunay là một đại diện lưới tốt hơn cho các đối tượng 2D có hình dạng tùy ý.

**- Định nghĩa 1:** Nếu D là một phân hoạch Delaunay, thì bất kỳ tam giác nào tn=(Pi,Pj,Pk)∈D đều thỏa mãn tính chất rằng đường tròn ngoại tiếp của tnt\_ntn​ không chứa bất kỳ điểm nút nào Pt​ trong nội bộ của nó.

Một lưới Delaunay cho một đối tượng video có thể được tạo ra qua các bước sau:

1. **Chọn các nút biên của lưới.** Một đa giác được sử dụng để xấp xỉ biên của đối tượng. Các đỉnh của đa giác này là các nút biên của lưới Delaunay. Một phép tiếp cận có thể là chọn các điểm biên có độ cong cao làm các nút biên.
2. **Chọn các nút nội bộ.** Các điểm đặc trưng bên trong biên của đối tượng như các điểm cạnh hoặc các góc có thể được chọn làm các nút nội bộ cho lưới.
3. **Thực hiện phân hoạch Delaunay.** Một phân hoạch Delaunay có ràng buộc được thực hiện trên các nút biên và nội bộ, với đa giác biên được sử dụng làm ràng buộc. Phân hoạch sẽ sử dụng các đoạn thẳng nối các nút biên liên tiếp làm cạnh và chỉ tạo các tam giác trong biên của đối tượng.

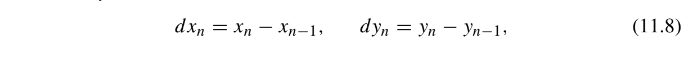
**-** **Phân hoạch Delaunay có ràng buộc.** Các cạnh nội bộ được thêm vào trước để tạo ra các tam giác mới. Thuật toán sẽ kiểm tra từng cạnh nội bộ để đảm bảo rằng nó là Delaunay tại chỗ. Đối với hai tam giác (Pi,Pj,Pk) và (Pj,Pk,Pl) chia sẻ một cạnh jkjkjk, nếu (Pi,Pj,Pk) chứa Pl hoặc (Pj,Pk,Pl) chứa Pi trong nội thất của đường tròn ngoại tiếp, thì cạnh jk không phải là Delaunay tại chỗ và sẽ bị thay thế bằng một cạnh mới il.

**-** Nếu Pl rơi chính xác trên đường tròn ngoại tiếp của tam giác (Pi,Pj,Pk) (và tương ứng, Pi​ cũng rơi chính xác trên đường tròn ngoại tiếp của tam giác (Pj,Pk,Pl)), thì cạnh jk sẽ được xem là Delaunay tại chỗ chỉ khi Pi​ hoặc Pl​ có hoành độ x lớn nhất trong số bốn nút.

- Hình 11.23a, b cho thấy tập hợp các nút lưới Delaunay và kết quả của phân hoạch Delaunay có ràng buộc. Nếu tổng số nút là N, và N=Nb+Ni ​trong đó Nb​ và Ni lần lượt là số nút biên và nút nội bộ, thì tổng số tam giác trong lưới Delaunay là Nb+2Ni−2. Trong hình trên, tổng số này là 8+2×6−2=18.

- Không giống như lưới đều, vị trí các nút trong lưới Delaunay không đều; do đó, chúng cần được mã hóa. Theo quy ước trong MPEG-4, vị trí (x0,y0) của nút biên trên cùng bên trái được mã hóa trước, tiếp theo là các điểm biên khác theo chiều ngược chiều kim đồng hồ (xem Hình 11.23a) hoặc theo chiều kim đồng hồ. Sau đó, vị trí của các nút bên trong được mã hóa theo bất kỳ thứ tự nào.

- Ngoại trừ vị trí đầu tiên (x0,y0) tất cả các tọa độ tiếp theo đều được mã hóa vi sai—nghĩa là, với n≥1.



A diagram of a hexagon and a hexagon with points

Description automatically generated

- Mã hóa chuyển động lưới 2D. Chuyển động của mỗi tam giác MOP trong lưới đều hoặc lưới Delaunay được mô tả bởi các vector chuyển động của ba nút đỉnh của nó. Một cấu trúc lưới mới chỉ có thể được tạo trong khung nội và cấu trúc tam giác của nó sẽ không thay đổi trong các khung liên tiếp sau đó. Điều này áp đặt việc ánh xạ một-một trong ước tính chuyển động lưới 2D.

- Với bất kỳ tam giác MOP (Pi,Pj,Pk) nào, nếu vector chuyển động của Pi và Pj​ lần lượt là MVi​ và MVj, thì một dự đoán Predk​ sẽ được thực hiện cho vector chuyển động của Pk​, được làm tròn đến độ chính xác nửa pixel:





- Khi ba vector chuyển động của tam giác MOP đầu tiên t0 đã được mã hóa, ít nhất một tam giác MOP lân cận sẽ chia sẻ một cạnh với t0​, và vector chuyển động cho nút đỉnh thứ ba của nó có thể được mã hóa, và cứ thế tiếp tục.

- Việc ước tính các vector chuyển động sẽ bắt đầu từ tam giác ban đầu t0​, là tam giác chứa nút biên trái trên cùng và nút biên tiếp theo theo chiều kim đồng hồ. Các vector chuyển động cho tất cả các nút khác trong MOP sẽ được mã hóa theo dạng vi sai, theo phương trình (11.10). Một thứ tự duyệt rộng (breadth-first) được thiết lập để đi qua các tam giác MOP trong quá trình mã hóa chuyển động lưới 2D.

- Hình 11.24 minh họa cách tạo cây bao trùm để thiết lập thứ tự duyệt rộng của các tam giác. Như đã thấy, tam giác ban đầu t0​ có hai tam giác lân cận chưa được ghé thăm t1​ và t2​. Chúng trở thành các nút con của t0 trong cây bao trùm.

- Tiếp theo, t1​ và t2​ có các tam giác lân cận chưa được ghé thăm (và do đó là các nút con) t3​, t4​ và t5​, t6​, tương ứng. Thứ tự duyệt hiện tại là t0​, t1​, t2​, t3​, t4​, t5​, theo thứ tự rộng. Một cấp dưới trong cây bao trùm, t3​ chỉ có một nút con là t7​ vì nút lân cận khác t1​ đã được ghé thăm; t4​ cũng chỉ có một nút con là t8​; và cứ như vậy.

**- Hoạt ảnh Đối tượng 2D:** Quá trình mã hóa chuyển động lưới trên đã thiết lập ánh xạ một-một giữa các tam giác lưới trong MOP tham chiếu và MOP đích. Nó tạo ra các vector chuyển động cho tất cả các điểm nút trong lưới 2D. Phép biến hình dựa trên lưới hiện được sử dụng để tạo kết cấu cho bề mặt hoạt hình mới bằng cách biến dạng kết cấu của từng tam giác trong MOP tham chiếu sang tam giác tương ứng trong MOP đích. Điều này hỗ trợ hoạt hình của các đối tượng video 2D tổng hợp.

- Đối với lưới tam giác, một hàm ánh xạ phổ biến cho phép biến dạng là phép biến đổi affine, vì nó biến đường thẳng thành đường thẳng và có thể đảm bảo tam giác được biến thành tam giác. Như sẽ được trình bày bên dưới, khi đã có sáu đỉnh của hai tam giác khớp nhau, ta có thể tìm ra các tham số cho phép biến đổi affine, nhờ đó phép biến đổi có thể được áp dụng cho tất cả các điểm bên trong tam giác đích để ánh xạ kết cấu.

- Cho một điểm P=(x,y) trên mặt phẳng 2D, có thể xác định một phép biến đổi tuyến tính sao cho



- Một phép biến đổi T là tuyến tính nếu T(αX+βY)=αT(X)+βT(Y), trong đó α và β là các hằng số. Phép biến đổi tuyến tính ở trên phù hợp cho các phép toán hình học như xoay và co giãn nhưng không phù hợp cho phép tịnh tiến, vì không thể cộng thêm một vector hằng số.

- Định nghĩa 2: Một phép biến đổi A là một phép biến đổi affine nếu và chỉ nếu tồn tại một vector C và một phép biến đổi tuyến tính T sao cho A(X)=T(X)+C.

- Nếu điểm (x,y) được biểu diễn dưới dạng [x,y,1] trong hệ tọa độ đồng nhất thường dùng trong đồ họa [24], thì một phép biến đổi affine chuyển [x,y,1] thành [x′,y′,1] được định nghĩa như sau:

A number on a white background

Description automatically generated

Nó thực hiện phép ánh xạ sau đây:

A close-up of numbers

Description automatically generated và có tối đa 6 bậc tự do được biểu diễn bởi các tham số a11,a21,a31,a12,a22và a32​.

- Các ma trận 3×3 sau đây là các phép biến đổi affine cho phép dịch chuyển theo (Tx,Ty), quay ngược chiều kim đồng hồ một góc θ, và tỉ lệ theo các hệ số Sx​ và Sy ​:

A group of black letters

Description automatically generated

- Sau đây là các phép biến đổi affine cho phép biến dạng theo trục x và trục y, tương ứng:

A black text with numbers

Description automatically generated with medium confidence

trong đó, Hx và Hy là các hằng số xác định mức độ của sự biến dạng (shear).

- Các phép biến đổi affine đơn giản ở trên có thể được kết hợp lại (thông qua phép nhân ma trận) để tạo ra các phép biến đổi affine tổng hợp — ví dụ, một phép dịch chuyển theo sau là phép xoay, hoặc một phép biến dạng (shear) theo sau là các phép biến đổi khác.

- Có thể chứng minh (xem Bài tập 11.6) rằng bất kỳ phép biến đổi tổng hợp nào được tạo ra theo cách này sẽ có dạng ma trận giống hệt nhau và có tối đa 6 bậc tự do, được xác định bởi các tham số a11, a21, a31, a12, a22 và a32.

- Nếu tam giác trong MOP mục tiêu là:



và tam giác tương ứng trong MOP tham chiếu là:



thì phép biến đổi giữa hai tam giác này có thể được xác định duy nhất bởi các công thức sau:

A black text on a white background

Description automatically generated

- Phương trình (11.15) chứa sáu phương trình tuyến tính (ba phương trình cho x≤ và ba phương trình cho y≤) cần thiết để giải quyết sáu hệ số ẩn a11, a21, a31, a12, a22, a32. Giả sử phương trình (11.15) có dạng X≤=XA. Ta có thể xác định ma trận AAA bằng cách sử dụng công thức A=X^(−1)X≤, trong đó ma trận nghịch đảo của X được tính theo công thức X^(−1)=adj(X) / det(X)’ trong đó adj(X) là ma trận phụ hợp của X và det(X) là định thức của X. Vì vậy, ta có công thức: det(X)=x0​(y1​−y2​)−y0​(x1​−x2​)+(x1​y2​−x2​y1​).

A close-up of a logo

Description automatically generated

A math equations in a box

Description automatically generated with medium confidence

- Vì ba đỉnh của tam giác lưới không bao giờ là những điểm đồng tuyến, ta có thể đảm bảo rằng ma trận X không suy biến, tức là det(X)≠0. Do đó, phương trình (11.16) luôn có nghiệm duy nhất.

- Phép biến đổi affine trên là phân mảnh, tức là mỗi tam giác có thể có phép biến đổi affine riêng của nó. Phương pháp này hoạt động tốt khi vật thể bị biến dạng nhẹ trong suốt quá trình hoạt hình. Hình 11.25a cho thấy một lưới Delaunay với một từ đơn giản được chiếu lên trên đó. Hình 11.25b cho thấy từ đã bị biến dạng trong một MOP tiếp theo trong chuỗi hoạt hình sau khi áp dụng phép biến đổi affine.

**Mã hóa dựa trên mô hình 3D**

- Do sự xuất hiện thường xuyên của khuôn mặt và cơ thể người trong video, MPEG-4 đã định nghĩa các mô hình 3D đặc biệt cho đối tượng khuôn mặt và đối tượng cơ thể. Một số ứng dụng tiềm năng cho các đối tượng video mới này bao gồm hội nghị truyền hình, giao diện người–máy tính, trò chơi và thương mại điện tử. Trong quá khứ, các mô hình khung dây 3D và các hoạt động của chúng đã được nghiên cứu cho hoạt hình đối tượng 3D [25]. MPEG-4 đi xa hơn các khung dây, để bề mặt của các đối tượng khuôn mặt hoặc cơ thể có thể được tô màu hoặc ánh xạ kết cấu.

- Mã hóa và Hoạt hình Đối tượng Khuôn mặt. Các mô hình khuôn mặt cho từng khuôn mặt có thể được tạo ra thủ công hoặc tạo ra tự động thông qua các kỹ thuật nhận dạng khuôn mặt và nhận dạng mẫu. Tuy nhiên, việc tạo ra thủ công là phiền phức và không đủ, còn việc tạo ra tự động vẫn chưa được thực hiện một cách đáng tin cậy.

A face with points and lines

Description automatically generated with medium confidence

- MPEG-4 đã áp dụng một mô hình khuôn mặt mặc định tổng quát, được phát triển bởi Tổ chức Ngôn ngữ Mô hình Hình ảnh Thực tế Ảo (VRML) [26]. Các Tham số Hoạt hình Khuôn mặt (FAPs) có thể được chỉ định để đạt được các hoạt động mong muốn — các sự sai lệch so với khuôn mặt “trung tính” ban đầu. Ngoài ra, các Tham số Định nghĩa Khuôn mặt (FDPs) có thể được chỉ định để mô tả tốt hơn các khuôn mặt cá nhân. Hình 11.26 cho thấy các điểm đặc trưng cho FDPs. Các điểm đặc trưng có thể bị ảnh hưởng bởi hoạt hình (FAPs) được hiển thị dưới dạng các vòng tròn đầy; và những điểm không bị ảnh hưởng được hiển thị dưới dạng các vòng tròn rỗng.

- Sáu mươi tám FAPs được định nghĩa [22]: FAP 1 dành cho các âm vị và FAP 2 dành cho các biểu cảm khuôn mặt. Các âm vị mã hóa các chuyển động môi rất thực tế bằng cách mô hình hóa vị trí miệng hiện tại của người nói. Tất cả các FAPs khác dành cho các chuyển động có thể có của đầu, hàm, môi, mí mắt, nhãn cầu, lông mày, con ngươi, cằm, má, lưỡi, mũi, tai, v.v. Ví dụ, các biểu cảm bao gồm trung tính, vui vẻ, buồn bã, tức giận, sợ hãi, ghê tởm và ngạc nhiên. Mỗi biểu cảm được thể hiện bằng một bộ các đặc điểm—chẳng hạn, buồn bã được thể hiện qua mắt hơi nhắm, miệng thả lỏng và lông mày cong lên. Các FAPs cho chuyển động bao gồm đầu\_nghiêng, đầu\_quay, đầu\_lăn, mở\_hàm, đẩy\_hàm, di\_chuyển\_hàm, đẩy\_môi\_dưới, đẩy\_môi\_trên, v.v.

- Để nén, các FAPs được mã hóa sử dụng mã hóa dự đoán. Các dự đoán cho FAPs trong khung hình mục tiêu được thực hiện dựa trên các FAPs trong khung hình trước, và các lỗi dự đoán sau đó được mã hóa bằng mã hóa số học. DCT cũng có thể được sử dụng để cải thiện tỷ lệ nén, mặc dù nó được coi là tốn kém về mặt tính toán. Các FAPs cũng được lượng tử hóa, với các kích thước bước lượng tử khác nhau được áp dụng để tận dụng thực tế là một số FAPs (ví dụ, mở\_hàm) cần ít độ chính xác hơn các FAPs khác (ví dụ, đẩy\_môi\_trên).

- **Mã hóa và Hoạt hình Vật thể Cơ thể**. MPEG-4 Phiên bản 2 đã giới thiệu các vật thể cơ thể, là sự mở rộng tự nhiên của các vật thể khuôn mặt.

- Làm việc với Nhóm Hoạt hình Nhân hình (H-Anim) trong Tổ chức VRML, MPEG đã áp dụng một cơ thể người ảo mặc định với tư thế mặc định. Tư thế mặc định là đứng, với đôi chân hướng về phía trước, tay để hai bên, với lòng bàn tay hướng vào trong. Có 296 Tham số Hoạt hình Cơ thể (BAPs). Khi áp dụng cho bất kỳ cơ thể ảo chuẩn MPEG-4 nào, chúng sẽ tạo ra cùng một hoạt hình.

- Một số lượng lớn BAPs mô tả góc khớp nối các phần cơ thể khác nhau, bao gồm cột sống, vai, xương đòn, khuỷu tay, cổ tay, ngón tay, hông, đầu gối, mắt cá chân và ngón chân. Điều này tạo ra 186 bậc tự do cho cơ thể, 25 cho mỗi bàn tay riêng biệt. Hơn nữa, một số chuyển động của cơ thể có thể được chỉ định ở nhiều cấp độ chi tiết. Ví dụ, năm cấp độ khác nhau, hỗ trợ 9, 24, 42, 60 và 72 bậc tự do có thể được sử dụng cho cột sống, tùy thuộc vào độ phức tạp của hoạt hình.

- Đối với các cơ thể cụ thể, các Tham số Định nghĩa Cơ thể (BDPs) có thể được chỉ định cho kích thước cơ thể, hình học bề mặt cơ thể và, tùy chọn, kết cấu. Hình học bề mặt cơ thể sử dụng đại diện lưới đa giác 3D, bao gồm một tập hợp các bề mặt phẳng đa giác trong không gian 3D [24]. Đại diện lưới 3D rất phổ biến trong đồ họa máy tính để mô hình hóa bề mặt. Kết hợp với ánh xạ kết cấu, nó có thể tạo ra các hình ảnh kết xuất tốt (chân thực).

- Việc mã hóa BAPs tương tự như FAPs: lượng tử hóa và mã hóa dự đoán được sử dụng, và lỗi dự đoán sau đó được nén thêm bằng mã hóa số học.

**11.4.4. Các Phần, Hồ sơ và Mức độ của MPEG-4:**

- Tính đến nay, MPEG-4 đã có hơn 28 phần [9], và vẫn còn nhiều phần khác đang được phát triển. Nó không chỉ định nghĩa Video trong Phần 2 và Âm thanh trong Phần 3, mà còn các chủ đề chuyên biệt như Đồ họa, Hoạt hình, Âm nhạc, Mô tả Cảnh, Mô tả Đối tượng, Khung tích hợp đa phương tiện (DMIF), Phát trực tuyến, và Quản lý và Bảo vệ Quyền sở hữu trí tuệ (IPMP) trong các phần khác nhau. MPEG-4 Phần 10 là về Mã hóa Video Nâng cao (AVC), giống như tiêu chuẩn H.264 AVC của ITU-T.

- MPEG-4 Phần 2 định nghĩa hơn 20 Hồ sơ hình ảnh, ví dụ: Simple, Advanced Simple, Core, Main, Simple Studio, v.v. Các hồ sơ thường dùng bao gồm: Simple Profile (SP) và Advanced Simple Profile (ASP). Hồ sơ ASP đã được một số phần mềm mã hóa video phổ biến áp dụng như DivX, Nero Digital, và Quicktime 6. Phần mềm mã nguồn mở Xvid hỗ trợ cả SP và ASP.

- Để phục vụ cho các ứng dụng khác nhau, MPEG-4 Phần 2 cũng định nghĩa nhiều Mức độ trong mỗi hồ sơ, ví dụ, từ L0 đến L3 cho SP, và từ L0 đến L5 cho ASP. Nhìn chung, các Mức độ thấp trong những hồ sơ này hỗ trợ các định dạng video băng thông thấp (CIF, QCIF) và các ứng dụng như hội nghị video trên web; trong khi các Mức độ cao hỗ trợ video chất lượng cao hơn.

**11.5. MPEG-7:**

- Khi nội dung đa phương tiện ngày càng trở thành một phần không thể thiếu trong các ứng dụng khác nhau, việc tìm kiếm hiệu quả và hiệu suất cao trở thành một mối quan tâm chính. Vào tháng 10 năm 1996, nhóm MPEG đã bắt đầu phát triển một tiêu chuẩn quan trọng khác, MPEG-7, tiếp nối các tiêu chuẩn MPEG-1, MPEG-2 và MPEG-4.

- Một điểm chung giữa MPEG-4 và MPEG-7 là sự tập trung vào các đối tượng âm thanh-hình ảnh. Mục tiêu chính của MPEG-7 [27–29] là phục vụ nhu cầu tìm kiếm nội dung âm thanh-hình ảnh (hoặc tìm kiếm đối tượng âm thanh-hình ảnh) trong các ứng dụng như thư viện số. Tuy nhiên, MPEG-7 không chỉ giới hạn trong việc tìm kiếm, mà còn có thể áp dụng cho bất kỳ ứng dụng đa phương tiện nào liên quan đến việc tạo ra (tạo nội dung) và sử dụng (tiêu thụ nội dung) dữ liệu đa phương tiện. Không giống như MPEG-1, -2 và -4, MPEG-7 không phải là một tiêu chuẩn "mới" cho mã hóa video.

- MPEG-7 trở thành tiêu chuẩn quốc tế vào tháng 9 năm 2001. Tên chính thức của nó là Giao diện Mô tả Nội dung Đa phương tiện, được tài liệu hóa trong ISO/IEC 15938 [30]. Bảy phần đầu tiên của tiêu chuẩn này bao gồm Hệ thống, Ngôn ngữ Định nghĩa Mô tả, Hình ảnh, Âm thanh, Các Sơ đồ Mô tả Đa phương tiện, Phần mềm Tham chiếu và Sự Phù hợp và Kiểm tra. Từ năm 2002, các phần 8 đến 12 tiếp tục được phát triển, chủ yếu tập trung vào các hồ sơ và định dạng truy vấn.

- MPEG-7 hỗ trợ nhiều ứng dụng đa phương tiện. Dữ liệu của nó có thể bao gồm hình ảnh tĩnh, đồ họa, mô hình 3D, âm thanh, lời nói, video và thông tin tổng hợp (cách kết hợp các yếu tố này). Các yếu tố dữ liệu MPEG-7 này có thể được biểu diễn dưới dạng văn bản hoặc nhị phân, hoặc cả hai. Phần 1 (Hệ thống) xác định cú pháp của định dạng nhị phân cho dữ liệu MPEG-7 (BiM). Phần 2 (Ngôn ngữ Định nghĩa Mô tả) xác định cú pháp của định dạng văn bản sử dụng XML Schema làm ngôn ngữ chính. Một ánh xạ mất mát hai chiều không mất dữ liệu được định nghĩa giữa các biểu diễn văn bản và nhị phân.

- Hình 11.27 minh họa một số ứng dụng có thể hưởng lợi từ MPEG-7. Như hình vẽ, các tính năng được trích xuất và sử dụng để khởi tạo các mô tả MPEG-7. Sau đó, chúng được mã hóa bởi bộ mã hóa MPEG-7 và gửi đến phương tiện lưu trữ và truyền tải. Các công cụ tìm kiếm và truy vấn khác nhau đưa ra các yêu cầu tìm kiếm và duyệt web, tạo thành các hoạt động "kéo" trên Internet, trong khi các đại lý lọc bỏ rất nhiều tài liệu được "đẩy" vào thiết bị đầu cuối—các người dùng và/hoặc hệ thống máy tính và ứng dụng tiêu thụ dữ liệu.

- Để mô tả nội dung đa phương tiện, MPEG-7 đã phát triển các Bộ mô tả (D), Sơ đồ mô tả (DS) và Ngôn ngữ Định nghĩa Mô tả (DDL). Dưới đây là một số thuật ngữ quan trọng:

* **Tính năng**. Một đặc tính của dữ liệu
* **Bộ mô tả (D).** Định nghĩa (cú pháp và ý nghĩa) của tính năng
* **Sơ đồ mô tả (DS).** Đặc tả cấu trúc và mối quan hệ giữa các Ds và DSs (xem [31])
* **Mô tả**. Một tập hợp các Ds và DSs đã được khởi tạo, mô tả thông tin cấu trúc và khái niệm về nội dung, lưu trữ và sử dụng nội dung, v.v.
* **Ngôn ngữ Định nghĩa Mô tả (DDL).** Các quy tắc cú pháp để biểu diễn và kết hợp các DS và D (xem [32])

A diagram of a computer process

Description automatically generated

- MPEG-7 đã làm rõ [30] rằng phạm vi của nó là chuẩn hóa các Ds, DSs và DDL cho các mô tả. Cơ chế và quy trình tạo ra và tiêu thụ các mô tả này nằm ngoài phạm vi của MPEG-7. Những yếu tố này để mở cho sự đổi mới và cạnh tranh trong ngành công nghiệp, và quan trọng hơn, để chào đón sự xuất hiện của các công nghệ mới ngày càng cải tiến.

- Tương tự như Mô hình Mô phỏng (SM) trong video MPEG-1, Mô hình Kiểm tra (TM) trong video MPEG-2, và các Mô hình Xác minh (VMs) trong MPEG-4 (video, âm thanh, SNHC và hệ thống), MPEG-7 gọi mô hình làm việc của mình là Mô hình Thí nghiệm (XM)—một trò chơi chữ theo thứ tự chữ cái! XM cung cấp các mô tả về các công cụ để đánh giá các Ds, DSs và DDL, giúp các thí nghiệm và xác minh có thể được thực hiện và so sánh bởi nhiều bên độc lập trên toàn thế giới. Bộ thí nghiệm đầu tiên của các thí nghiệm này được gọi là các thí nghiệm lõi.

**11.5.1. Định nghĩa của đặc điểm (feature):**

- Các mô tả MPEG-7 được thiết kế để mô tả cả đặc điểm cấp thấp, như màu sắc, kết cấu, hình dạng và chuyển động, cũng như các đặc điểm cấp cao của các đối tượng ngữ nghĩa, như sự kiện và các khái niệm trừu tượng. Như đã đề cập trước đó, các phương pháp và quy trình cho việc trích xuất đặc điểm tự động hoặc bán tự động không phải là một phần của tiêu chuẩn. Mặc dù đã có những nỗ lực và tiến bộ trong các lĩnh vực xử lý hình ảnh và video, thị giác máy tính và nhận dạng mẫu, việc trích xuất đặc điểm tự động và đáng tin cậy vẫn không được kỳ vọng sẽ có trong tương lai gần, đặc biệt là ở cấp độ cao.

- Các mô tả được chọn dựa trên việc so sánh hiệu suất, hiệu quả và kích thước của chúng. Các mô tả hình ảnh cấp thấp cho các đặc điểm hình ảnh cơ bản bao gồm

• **Màu sắc**  
– **Không gian màu.** (a) RGB, (b) YCbCr, (c) HSV (màu sắc, độ bão hòa, giá trị) [24], (d) HMMD (HueMaxMinDiff) [34], (e) Không gian màu 3D có thể suy ra từ RGB qua ma trận 3 × 3, (f) đơn sắc  
– **Lượng tử hóa màu.** (a) Tuyến tính, (b) phi tuyến, (c) bảng tra cứu  
– **Màu sắc nổi bật.** Một số ít màu sắc đại diện trong mỗi vùng hoặc hình ảnh. Chúng hữu ích cho việc tìm kiếm hình ảnh dựa trên sự tương đồng về màu sắc  
– **Màu sắc có thể mở rộng.** Một biểu đồ màu trong không gian màu HSV. Nó được mã hóa bằng biến đổi Haar và do đó có thể mở rộng  
– **Bố cục màu sắc.** Phân phối không gian của các màu sắc cho việc tìm kiếm dựa trên bố cục màu sắc  
– **Cấu trúc màu sắc.** Tần suất của một phần tử cấu trúc màu mô tả cả nội dung màu sắc và cấu trúc của nó trong hình ảnh. Phần tử cấu trúc màu được tạo thành từ vài mẫu hình ảnh trong vùng lân cận có cùng màu sắc  
– **Màu sắc nhóm khung/Giới nhóm hình ảnh (GoF/GoP).** Tương tự như màu sắc có thể mở rộng, ngoại trừ việc áp dụng cho một đoạn video hoặc một nhóm hình ảnh tĩnh. Một biểu đồ màu tổng hợp được thu được bằng cách áp dụng các phép toán trung bình, trung vị, hoặc giao nhau lên các nhóm phân đoạn của các biểu đồ màu trong GoF/GoP và sau đó được gửi đến biến đổi Haar

• **Kết cấu (Texture)**  
– **Kết cấu đồng nhất.** Sử dụng bộ lọc Gabor điều chỉnh theo hướng và tỉ lệ [35] để đại diện một cách định lượng các vùng kết cấu đồng nhất. Ưu điểm của bộ lọc Gabor là chúng cung cấp độ phân giải tối ưu đồng thời trong cả miền không gian và tần số không gian [36]. Ngoài ra, chúng là bộ lọc dải băng phù hợp với đặc điểm thị giác của con người. Một bộ lọc bao gồm 30 bộ lọc Gabor, với năm tỉ lệ khác nhau và sáu hướng khác nhau cho mỗi tỉ lệ, được sử dụng để trích xuất mô tả kết cấu  
– **Duyệt kết cấu.** Mô tả sự đều đặn, độ thô và tính hướng của các cạnh được sử dụng để đại diện và duyệt các kết cấu đồng nhất. Một lần nữa, bộ lọc Gabor được sử dụng  
– **Biểu đồ cạnh.** Đại diện phân phối không gian của bốn cạnh có hướng (0◦, 45◦, 90◦, 135◦) và một cạnh không hướng. Hình ảnh được chia thành các hình ảnh con nhỏ, và một biểu đồ cạnh với năm ô chứa được tạo ra cho mỗi hình ảnh con.

• **Hình dạng (Shape)**

– **Hình dạng dựa trên vùng (Region-based shape).** Một tập hợp các hệ số Biến đổi Bán kính Góc (Angular Radial Transform - ART) [28] được sử dụng để mô tả hình dạng của một đối tượng. Một đối tượng có thể bao gồm một hoặc nhiều vùng, có thể có các lỗ trong đối tượng. Biến đổi ART là một biến đổi phức tạp 2D được định nghĩa trong hệ tọa độ cực trên một đĩa đơn vị. Các hàm cơ sở ART có thể phân tách theo chiều góc và chiều bán kính. Ba mươi sáu hàm cơ bản, 12 theo chiều góc và 3 theo chiều bán kính, được sử dụng để trích xuất mô tả hình dạng.  
– **Hình dạng dựa trên đường viền (Contour-based shape).** Sử dụng biểu diễn không gian độ cong (Curvature Scale Space - CSS) [37] không thay đổi theo tỷ lệ và xoay, và mạnh mẽ trước chuyển động không đàn hồi và sự che khuất một phần của hình dạng.  
– **Hình dạng 3D.** Mô tả các mô hình lưới 3D và chỉ số hình dạng [38]. Biểu đồ các chỉ số hình dạng trên toàn bộ lưới được sử dụng làm mô tả.

A diagram of a machine

Description automatically generated

• **Chuyển động (Motion)**

– **Chuyển động của máy quay (Camera motion).** Các kiểu chuyển động như cố định, quét ngang (pan), nghiêng (tilt), xoay (roll), di chuyển ngang (dolly), theo dõi (track), và cẩu (boom) (Xem Hình 11.28 và [39]).  
– **Quỹ đạo chuyển động của đối tượng (Object motion trajectory).** Một danh sách các điểm chính (x, y, z, t). Các hàm nội suy tùy chọn được sử dụng để xác định gia tốc dọc theo đường đi (Xem [39]).  
– **Chuyển động đối tượng tham số (Parametric object motion).** Mô hình cơ bản là mô hình affine 2D cho phép dịch chuyển, xoay, tỷ lệ, cắt xén, và sự kết hợp của các yếu tố này. Mô hình phối cảnh phẳng và mô hình bậc hai có thể được sử dụng cho sự biến dạng phối cảnh và chuyển động phức tạp hơn.  
– **Hoạt động chuyển động (Motion activity).** Cung cấp các mô tả về cường độ, nhịp độ, cảm xúc, v.v., của video—ví dụ, "ghi điểm trong một trận đấu khúc côn cầu" hoặc "phỏng vấn một người."

• **Định vị (Localization)**

– **Định vị vùng (Region locator).** Xác định vị trí của các vùng trong hình ảnh bằng một hộp hoặc đa giác.  
– **Định vị không gian-thời gian (Spatiotemporal locator).** Mô tả các vùng không gian-thời gian trong các chuỗi video. Sử dụng một hoặc nhiều bộ mô tả của các vùng và chuyển động của chúng.

• **Khác (Others)**

– **Nhận diện khuôn mặt (Face recognition).** Hình ảnh khuôn mặt đã chuẩn hóa được đại diện dưới dạng một vector 1D, sau đó chiếu lên một bộ 49 vector cơ sở, đại diện cho tất cả các vector khuôn mặt có thể.

**11.5.2. Description Scheme (DS):**

- MPEG-7 Description Schemes (DSs) tổng quan trong các lĩnh vực sau:

* **Yếu tố cơ bản**:
  + **Kiểu dữ liệu và cấu trúc toán học**: Bao gồm vector, ma trận, histogram và các yếu tố liên quan.
  + **Cấu trúc**: Liên kết các tệp media, xác định các đoạn, vùng và các thành phần khác.
  + **Công cụ sơ đồ**: Bao gồm các phần tử gốc (điểm bắt đầu của các tài liệu XML MPEG-7), các phần tử cấp cao (để tổ chức các DS theo nội dung cụ thể), và công cụ đóng gói (nhóm các thành phần DS có liên quan thành các gói).
* **Quản lý nội dung**:
  + **Mô tả phương tiện**: Bao gồm một DS duy nhất, MediaInformation DS, chứa MediaIdentification D và một hoặc nhiều Media Profile Ds cung cấp thông tin như phương thức mã hóa, gợi ý chuyển mã và các định dạng lưu trữ/đưa ra.
  + **Mô tả tạo dựng và sản xuất**: Bao gồm thông tin liên quan đến việc tạo ra (tiêu đề, người tạo, địa điểm, ngày tạo), phân loại (thể loại, ngôn ngữ, hướng dẫn độ tuổi), và tài liệu liên quan.
  + **Mô tả sử dụng nội dung**: Các DS khác nhau cung cấp thông tin về quyền sử dụng, hồ sơ sử dụng, tính khả dụng và các yếu tố tài chính (chi phí sản xuất, thu nhập từ việc sử dụng nội dung).
* **Mô tả Nội dung**
  + **Mô tả Cấu trúc.** Một Segment DS mô tả các khía cạnh cấu trúc của nội dung. Một đoạn (segment) là một phần của đối tượng nghe nhìn. Mối quan hệ giữa các đoạn thường được biểu diễn dưới dạng cây đoạn. Khi mối quan hệ không hoàn toàn có tính chất phân cấp, một đồ thị đoạn được sử dụng.
    - Segment DS có thể được triển khai dưới dạng một đối tượng lớp (class object). Nó có năm lớp con: Audiovisual Segment DS, Audio Segment DS, Still Region DS, Moving Region DS, và Video Segment DS. Các lớp con DS này có thể đệ quy có các lớp con riêng của chúng.
    - Ví dụ, một Still Region DS có thể được dùng để mô tả một hình ảnh theo các khía cạnh như quá trình tạo ra (tiêu đề, tác giả, ngày tạo), sử dụng (bản quyền), phương tiện (định dạng tệp), chú thích văn bản, biểu đồ màu, và có thể cả các mô tả kết cấu, v.v. Khu vực ban đầu (trong trường hợp này là hình ảnh) có thể được chia nhỏ thành nhiều vùng, và mỗi vùng này lại có thể có các DS riêng của chúng.
    - Hình 11.29 minh họa một Đoạn video cho một nhiệm vụ cứu hộ trên biển, trong đó một người được hạ xuống thuyền từ một chiếc trực thăng. Có ba vùng chuyển động bên trong Đoạn video. Một đồ thị đoạn có thể được xây dựng để bao gồm các mô tả cấu trúc như thành phần của khung hình video (trực thăng, người, thuyền), mối quan hệ không gian và chuyển động (phía trên, trên, gần với, di chuyển đến, v.v.) của các vùng này.
  + **Mô tả Khái niệm.** Phần này bao gồm mô tả ở cấp độ cao hơn (không thuộc cấu trúc) về nội dung, chẳng hạn như Event DS cho trận bóng rổ hoặc trận đấu của đội Lakers, Object DS cho các đối tượng như "John" hoặc "người", State DS để chỉ các thuộc tính ngữ nghĩa tại một thời điểm hoặc địa điểm cụ thể, và Concept DS cho các khái niệm trừu tượng như "tự do" hay "bí ẩn".
    - Tương tự như Segment DS, các khái niệm DS cũng có thể được tổ chức dưới dạng cây hoặc đồ thị.

A collage of several helicopters

Description automatically generated

• **Dẫn hướng và truy cập**

– **Tóm tắt.** Phần này cung cấp tóm tắt video để hỗ trợ duyệt và dẫn hướng nhanh chóng qua nội dung, thường bằng cách hiển thị các khung hình chính. Các DS được hỗ trợ bao gồm: Summarization DS, HierarchicalSummary DS, HighlightLevel DS, và SequentialSummary DS. Tóm tắt theo cấu trúc phân cấp cung cấp một hệ thống phân cấp các khung hình chính ở nhiều cấp độ, trong khi tóm tắt tuần tự thường trình bày dưới dạng trình chiếu hoặc lướt qua nghe nhìn, có thể kèm theo âm thanh và văn bản đồng bộ.

Hình 11.30 minh họa một tóm tắt cho video về cuộc diễu hành và đua thuyền rồng trong công viên. Tóm tắt này được tổ chức thành cấu trúc phân cấp ba cấp độ. Mỗi đoạn video ở mỗi cấp độ được thể hiện bằng một khung hình chính có kích thước hình thu nhỏ.

– **Phân chia và Phân giải.** Phần này liên quan đến phân chia và phân giải hình ảnh hoặc video. Các phân chia View (được chỉ định bởi View DSs) mô tả các góc nhìn khác nhau về không gian và tần số của dữ liệu nghe nhìn, chẳng hạn như góc nhìn không gian (có thể là một đoạn không gian của hình ảnh), góc nhìn thời gian (như trong một đoạn thời gian của video), góc nhìn tần số (như trong dải con wavelet của hình ảnh), hoặc góc nhìn độ phân giải (như trong một hình ảnh thu nhỏ), v.v.

Các phân giải View DSs xác định các phân giải cây hoặc đồ thị khác nhau để tổ chức các góc nhìn của dữ liệu nghe nhìn, chẳng hạn như một SpaceTree DS (phân giải hình ảnh theo cây quad).

– **Biến thể của nội dung.** Một Variation DS chỉ định một biến thể từ dữ liệu gốc như độ phân giải hình ảnh, tốc độ khung hình, giảm màu, nén, v.v. DS này có thể được các máy chủ sử dụng để điều chỉnh việc truyền tải dữ liệu nghe nhìn phù hợp với các đặc điểm mạng và thiết bị đầu cuối cho một chất lượng dịch vụ (QoS) nhất định.

• **Tổ chức Nội dung**

– **Bộ sưu tập.** CollectionStructure DS tập hợp các nội dung nghe nhìn thành các cụm. Nó xác định các thuộc tính chung của các phần tử trong cụm và các mối quan hệ giữa các cụm.

– **Mô hình.** Model DSs bao gồm Probability model DS, Analytic model DS, và Classifier DS, dùng để trích xuất các mô hình và thống kê các thuộc tính và đặc điểm của các bộ sưu tập.

A diagram of a structure

Description automatically generated with medium confidence

• **Tương tác Người dùng**

– **Ưu tiên Người dùng.** DSs mô tả các sở thích của người dùng trong việc tiêu thụ nội dung nghe nhìn, chẳng hạn như các loại nội dung, chế độ duyệt, đặc điểm riêng tư và khả năng điều chỉnh sở thích bởi một tác nhân phân tích hành vi của người dùng.

**11.5.3. Ngôn ngữ Định nghĩa Mô tả (DDL):**

- MPEG-7 đã chọn Ngôn ngữ Lược đồ XML (XML Schema Language) do Tổ chức World Wide Web (W3C) phát triển ban đầu làm Ngôn ngữ Định nghĩa Mô tả (DDL) của nó. Vì XML Schema Language không được thiết kế đặc biệt cho nội dung nghe nhìn, nên một số phần mở rộng đã được thực hiện. Không đi vào chi tiết cụ thể, MPEG-7 DDL có các thành phần sau:

• **Các thành phần cấu trúc của XML Schema**

– **Schema** — lớp bao quanh các định nghĩa và khai báo.  
– **Các thành phần cấu trúc chính**, chẳng hạn như định nghĩa kiểu dữ liệu đơn giản và phức tạp, và khai báo thuộc tính và phần tử.  
– **Các thành phần cấu trúc phụ**, chẳng hạn như định nghĩa nhóm thuộc tính, định nghĩa ràng buộc danh tính, định nghĩa nhóm và khai báo ký hiệu.  
– **Các thành phần "hỗ trợ"**, như chú thích, phần tử và ký tự đại diện.

• **Các thành phần kiểu dữ liệu của XML Schema**

– Kiểu dữ liệu nguyên thủy và dẫn xuất  
– Cơ chế cho phép người dùng tạo ra kiểu dữ liệu mới  
– Kiểm tra kiểu dữ liệu tốt hơn XML 1.0

• **Phần mở rộng của MPEG-7**

– Kiểu dữ liệu mảng và ma trận  
– Nhiều loại phương tiện, bao gồm âm thanh, video và các trình chiếu nghe nhìn  
– Các kiểu dữ liệu liệt kê cho MimeType, CountryCode, RegionCode, CurrencyCode và CharacterSetCode  
– Quản lý và Bảo vệ Sở hữu trí tuệ (IPMP) cho Ds và DSs

**11.6. Bài tập:**

1. Như chúng ta đã biết, nén video MPEG sử dụng các khung I-, P-, và B-frames. Tuy nhiên, tiêu chuẩn H.261 trước đây không sử dụng B-frames. Hãy mô tả một tình huống mà việc nén video sẽ không hiệu quả nếu không có B-frames (Câu trả lời của bạn nên khác với câu trả lời trong Hình 11.1).
2. Tiêu chuẩn MPEG-1 giới thiệu B-frames, và phạm vi tìm kiếm vector chuyển động đã được tăng từ [−15, 15] trong H.261 lên [−512, 511.5]. Tại sao việc này lại cần thiết? Tính toán số lượng B-frames giữa các P-frames liên tiếp sẽ biện minh cho sự tăng này.
3. B-frames mang lại những lợi ích rõ ràng trong việc mã hóa, chẳng hạn như tăng SNR ở bitrate thấp và tiết kiệm băng thông. Vậy nhược điểm của B-frames là gì?
4. Vẽ lại Hình 11.8 của bộ mã hóa và giải mã SNR mở rộng hai lớp MPEG-2 để bao gồm một lớp nâng cao thứ hai.
5. Vẽ sơ đồ khối cho bộ mã hóa và giải mã MPEG-2 cho (a) tính năng mở rộng SNR và không gian, (b) tính năng mở rộng SNR và thời gian.
6. Tại sao B-frames không được sử dụng làm khung tham chiếu cho bù chuyển động? Giả sử có một chế độ trong đó bất kỳ loại khung hình nào cũng có thể được chỉ định làm khung tham chiếu. Thảo luận về các sự đánh đổi khi sử dụng B-frames làm khung tham chiếu thay vì P-frames trong một chuỗi video (tức là loại bỏ hoàn toàn P-frames).
7. Viết một chương trình để triển khai tính năng mở rộng SNR trong MPEG-2. Chương trình của bạn phải có khả năng hoạt động trên bất kỳ macroblock nào sử dụng bất kỳ kích thước bước lượng tử nào và phải xuất ra cả hai bitstream Bits\_base và Bits\_enhance. Bước mã hóa độ dài biến có thể bỏ qua.
8. MPEG-4 bù chuyển động được cho là dựa trên VOP. Cuối cùng, VOP vẫn được chia thành các macroblock (macroblock nội bộ, macroblock biên, v.v.) để thực hiện bù chuyển động.

(a) Những vấn đề tiềm ẩn của việc triển khai hiện tại là gì? Làm thế nào để cải thiện chúng?  
(b) Liệu có thể có bù chuyển động thực sự dựa trên VOP không? Nó sẽ so sánh như thế nào với việc triển khai hiện tại?

1. MPEG-1, 2, và 4 đều được biết đến như là các tiêu chuẩn giải mã. Các thuật toán nén, do đó, các chi tiết của bộ mã hóa, được để mở cho các cải tiến và phát triển trong tương lai. Đối với MPEG-4, vấn đề chính của phân đoạn đối tượng video—làm thế nào để có được các VOP—được để không xác định.

(a) Đề xuất một số phương pháp của bạn cho phân đoạn đối tượng video.  
(b) Những vấn đề tiềm ẩn của phương pháp của bạn là gì?

1. Các vector chuyển động có thể có độ chính xác dưới điểm ảnh. Cụ thể, MPEG-4 cho phép độ chính xác một phần tư điểm ảnh trong các VOP độ sáng. Hãy mô tả một thuật toán sẽ thực hiện độ chính xác này.
2. Là một dự án lập trình, tính toán SA-DCT cho khối 8 × 8 sau:

A number grid with numbers

Description automatically generated

1. Chi phí tính toán của SA-DCT so với DCT thông thường là bao nhiêu? Giả sử đối tượng video là một hình vuông 4 × 4 ở giữa khối 8 × 8.
2. Các phép biến đổi affine có thể được kết hợp để tạo ra một phép biến đổi affine tổng hợp. Chứng minh rằng phép biến đổi tổng hợp sẽ có dạng ma trận giống hệt (với [001] T là cột cuối cùng) và tối đa có 6 bậc tự do, được chỉ định bởi các tham số a11, a21, a31, a12, a22, a32.
3. Mã hóa chuyển động dựa trên lưới hoạt động khá tốt cho hoạt hình 2D và hoạt hình khuôn mặt. Vậy vấn đề chính khi áp dụng nó vào hoạt hình cơ thể là gì?
4. Động lực chính đằng sau sự phát triển của MPEG-7 là gì? Hãy đưa ra ba ví dụ về các ứng dụng trong thế giới thực có thể hưởng lợi từ MPEG-7.
5. Hai trong số các mô tả hình dạng chính trong MPEG-7 là "dựa trên vùng" và "dựa trên đường viền". Dĩ nhiên, có rất nhiều cách để mô tả hình dạng của các vùng và đường viền.  
   (a) Mô tả hình dạng yêu thích của bạn là gì?  
   (b) Nó sẽ so sánh như thế nào với ART và CSS trong MPEG-7?

**Tài liệu tham khảo**

1. L. Chiariglione, The development of an integrated audiovisual coding standard: MPEG. Proc.

IEEE 83, 151–157 (1995)

2. D.J. Le Gall, MPEG: a video compression standard for multimedia applications. Commun.

ACM 34(4), 46–58 (1991)

3. R. Schafer, T. Sikora, Digital video coding standards and their role in video communications.

Proc. IEEE 83(6), 907–924 (1995)

4. Information technology—Coding of moving pictures and associted audio for digital storage

media at up to about 1.5 Mbit/s. Int. Standard: ISO/IEC 11172, Parts 1–5 (1992)

5. J.L. Mitchell, W.B. Pennebaker, C.E. Fogg, D.J. LeGall, MPEG Video Compression Standard.

(Chapman & Hall, New York, 1996)

6. B.G. Haskell, A. Puri, A. Netravali, Digital Video: an Introduction to MPEG-2. (Chapman &

Hall, New York, 1996)

7. Information technology—Generic coding of moving pictures and associated audio information.

Int. Standard: ISO/IEC 13818, Parts 1–11 (2004)

8. T. Sikora, The MPEG-4 video standard verification model. IEEE Trans. Circuits Syst. Video

Technol. (Special issue on MPEG-4) 7(1), 19–31 (1997)

9. Information technology—Generic coding of audio-visual objects. Int. Standard: ISO/IEC

14496, Parts 1–28 (2012)

10. A. Puri, T. Chen (eds.), Multimedia Systems, Standards, and Networks (Marcel Dekker, New

York, 2000)

11. G. Fernando et al., Java in MPEG-4 (MPEG-J), in Multimedia, Systems, Standards, and Net-

works, ed. by A. Puri, T. Chen (Marcel Dekker, New York, 2000), pp. 449–460

12. Video Coding for Low Bit Rate Communication, ITU-T Recommendation H.263, Version 1,

1995, Version 2, 1998, Version 3, 2000, Revised 2005

13. A. Puri et al., MPEG-4 natural video coding—Part I, in Multimedia, Systems, Standards, and

Networks, ed. by A. Puri, T. Chen (Marcel Dekker, New York, 2000), pp. 205–244

14. T. Ebrahimi, F. Dufaux, Y. Nakaya, MPEG-4 natural video coding - Part II, in Multimedia,

Systems, Standards, and Networks, ed. by A. Puri, T. Chen (Marcel Dekker, New York, 2000),

pp. 245–269

15. P. Kauff, et al. Functional coding of video using a shape-adaptive DCT algorithm and an object-

based motion prediction toolbox. IEEE Trans. Circuits Syst. Video Technol. (Special issue on

MPEG-4) 7(1), 181–196 (1997)

16. Standardization of Group 3 facsimile apparatus for document transmission. ITU-T

Recommendation T.4, 1980

17. Facsimile coding schemes and coding control functions for Group 4 facsimile apparatus. ITU-T

Recommendation T.6, 1984

18. Information technology—Coded representation of picture and audio information—progressive

bi-Level image compression. Int. Standard: ISO/IEC 11544, also ITU-T Recommendation T.82,

1992

19. J.M. Shapiro, Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans.

Signal Process. 41(12), 3445–3462 (1993)

20. G. Wolberg, Digital Image Warping (Computer Society Press, Los Alamitos, CA, 1990)

21. M.C. Lee, et al. A layered video object coding system using sprite and affine motion model.

IEEE Trans. Circuits Syst. Video Technol. 7(1), 130–145 (1997)

22. P. van Beek, MPEG-4 synthetic video, in Multimedia, Systems, Standards, and Networks, ed.

by A. Puri, T. Chen (Marcel Dekker, New York, 2000), pp. 299–330

23. A.M. Tekalp, P. van Beek, C. Toklu, B. Gunsel, 2D mesh-based visual object representation

for interactive synthetic/natural digital video. Proc. IEEE 86, 1029–1051 (1998)

24. John F. Hughes, Andries van Dam, Morgan McGuire, David F. Sklar, James D. Foley, Steven

K. Feiner, K. Akeley, Computer Graphics: Principles and Practice, 3rd ed. (Addison-Wesley,

Upper Saddle River, 2013)

25. A. Watt, M. Watt, Advanced Animation and Rendering Techniques. (Addison-Wesley, Upper

Saddle River, 1992)

26. Information technology—The Virtual Reality Modeling Language—Part 1: Functional speci-

fication and UTF-8 encoding. Int. Standard: ISO/IEC 14772–1 (1997)

27. S.F. Chang, T. Sikora, A. Puri, Overview of the MPEG-7 standard. IEEE Trans. Circuits Syst.

Video Technol. (Special issue on MPEG-7) 11(6), 688–695 (2001)

28. B.S. Manjunath, P. Salembier, T. Sikora (eds.), Introduction to MPEG-7: Multimedia Content

Description Interface. (Wiley, Chichester, 2002)

29. H.G. Kim, N. Moreau, T. Sikora, MPEG-7 Audio and Beyond: Audio Content Indexing and

Retrieval. (Wiley, New York, 2005)

30. Information technology—Multimedia content description interface. Int. Standard: ISO/IEC

15938, Parts 1–12 (2008)

31. P. Salembier, J. R. Smith, MPEG-7 multimedia description schemes. IEEE Trans. Circuits Syst.

Video Technol. 11(6), 748–759 (2001)

32. J. Hunter, F. Nack, An overview of the MPEG-7 description definition language (DDL) pro-

posals. Signal Process. Image Commun. 16(1–2), 271–293 (2001)

33. T. Sikora, The MPEG-7 visual standard for content description—an overview. IEEE Trans.

Circuits Syst. Video Technol. (Special issue on MPEG-7) 11(6), 696–702 (2001)

34. B.S. Manjunath, J.-R. Ohm, V.V. Vasudevan, A. Yamada, Color and texture descriptors. IEEE

Trans. Circuits Syst. Video Technol. 11, 703–715 (2001)

35. B.S. Manjunath, G.M. Haley, W.Y. Ma, in Multiband Techniques for Texture Classification and

Segmentation, ed. by A. Bovik, Handbook of Image and Video Processing (Academic Press,

San Diego, 2000), pp. 367–381

36. T.P. Weldon, W.E. Higgins, D.F. Dunn, Efficient Gabor filter design for texture segmentation.

Pattern Recogn 29(12), 2005-2015 (1996)

37. F. Mokhtarian, A.K. Mackworth, A theory of multiscale, curvature-based shape representation

for planar curves. IEEE Trans. Pattern Anal. Mach. Intell. 14(8), 789–805 (1992)

38. J.J. Koenderink, A.J. van Doorn, Surface shape and curvature scales. Image Vision Comput.

10, 557–565 (1992)

39. S. Jeannin et al., Motion descriptor for content-based video representation. Signal Process.

Image Commun. 16(1–2), 59–85 (2000)