

Codificació

Alexandre Ribelles García

PID_00198470



Els textos i imatges publicats en aquesta obra estan subjectes –llevat que s'indiqui el contrari– a una llicència de Reconeixement-NoComercial-SenseObraDerivada (BY-NC-ND) v.3.0 Espanya de Creative Commons. Podeu copiar-los, distribuir-los i transmetre'ls públicament sempre que en citeu l'autor i la font (FUOC. Fundació per a la Universitat Oberta de Catalunya), no en feu un ús comercial i no en feu obra derivada. La llicència completa es pot consultar a <http://creativecommons.org/licenses/by-nc-nd/3.0/es/legalcode.ca>

Índex

| | |
|--|----|
| Introducció | 5 |
| Objectius | 6 |
| 1. Conceptes de codificació | 7 |
| 1.1. Codificació i compressió | 9 |
| 1.2. Redundància i entropia | 11 |
| 1.3. Mesura de la compressió d'imatge | 12 |
| 1.4. Mesura objectiva i subjectiva de la qualitat d'imatge | 13 |
| 1.5. Classificació de les tècniques de compressió | 14 |
| 1.5.1. Compressió sense pèrdues | 14 |
| 1.5.2. Compressió amb pèrdues | 15 |
| 1.6. Tractament del color: representació i submostratge | 17 |
| 1.6.1. Representacions del color: espais RGB i YCbCr | 18 |
| 1.6.2. Submostratge de color | 20 |
| 1.7. Exercicis | 22 |
| 2. Codificació d'imatge fixa | 24 |
| 2.1. El format JPEG | 24 |
| 2.1.1. Mode seqüencial | 25 |
| 2.1.2. Altres modes de treball de JPEG | 33 |
| 2.2. JPEG 2000 | 35 |
| 2.3. JPEG XR | 36 |
| 2.4. Tendències futures de la codificació d'imatge | 36 |
| 2.5. Exercicis | 37 |
| 3. Codificació de vídeo i àudio | 45 |
| 3.1. Models de codificació d'àudio i vídeo | 45 |
| 3.2. Compressió d'àudio digital | 47 |
| 3.3. Diferenciar contenidor i còdec | 49 |
| 4. L'estàndard MPEG-2 | 52 |
| 4.1. Antecedents | 52 |
| 4.2. Què és MPEG-2 | 53 |
| 4.3. Conceptes bàsics: GOP i MPEG | 54 |
| 4.3.1. GOP | 54 |
| 4.3.2. MPEG | 57 |
| 4.3.3. Exemples d'ús | 59 |
| 4.4. Els fluxos elementals de dades | 60 |
| 4.4.1. Flux elemental de vídeo | 60 |
| 4.4.2. Flux elemental d'àudio | 62 |

| | |
|---|-----------|
| 4.4.3. Flux elemental empaquetat | 62 |
| 4.5. Emmagatzematge i transmissió | 63 |
| 4.5.1. Emmagatzematge | 63 |
| 4.5.2. Transmissió | 63 |
| 4.6. Perfils i nivells en MPEG-2 | 64 |
| 4.6.1. Perfils | 64 |
| 4.6.2. Nivells | 64 |
| 4.6.3. Combinacions de perfils i nivells | 65 |
| 4.7. MPEG-2 en la indústria | 65 |
| 4.8. Mesura de la qualitat d'imatge en MPEG-2 per a DVB | 66 |
| 5. L'estàndard MPEG-4 i H.264/AVC..... | 69 |
| 5.1. L'estàndard i les patents | 69 |
| 5.2. H.264 o MPEG-4 Part10 | 70 |
| 5.3. Emmagatzematge | 71 |
| 5.4. Transmissió | 71 |
| 5.5. Perfils i nivells | 72 |
| 5.5.1. Perfils | 72 |
| 5.5.2. Nivells | 72 |
| 5.6. H.264 en la indústria | 72 |
| 5.7. Gestió d'objectes multimèdia amb MPEG-4 | 74 |

Introducció

En aquest mòdul iniciem una visió detallada dels estàndards de codificació d'imatge, so i vídeo. Per a fer-ho, discutirem la necessitat d'aquests estàndards, introduïrem els conceptes de *redundància* i *entropia*, *espai* i *submostratge de color*, descriurem els tipus de compressió i proporcionarem les eines empíriques i subjectives per a valorar la compressió.

El primer estàndard de codificació que analitzarem, JPEG per a imatge, incorpora elements que formen part de la majoria de sistemes de codificació, de manera que el descriurem detalladament tant des del punt de vista pràctic com teòric. Després en compararem els modes de treball, amb pèrdues i sense, i n'introduïrem les versions més actuals de l'estàndard: 2000 i XR.

Tot seguit, augmentarem de complexitat i ens endinsarem en la codificació d'àudio i vídeo mitjançant l'estàndard MPEG-2, introduint els tipus i les estructures d'imatge i també la descripció del codificador (*coder*), el descodificador (*decoder*) i els modes de treball per perfils i nivells.

Finalment, tancarem el mòdul amb l'estàndard H.264/AVC, anomenat col·loquialment *MPEG-4*, tot fent-ne una comparativa amb l'estàndard que el precedia i descrivint-ne les novetats i aplicacions actuals segons els seus perfils i nivells.

Objectius

Els objectius d'aquest mòdul són els següents:

- 1.** Identificar els paràmetres que caracteritzen la imatge i el so.
- 2.** Diferenciar els conceptes de *codificació* i *compressió*.
- 3.** Identificar i diferenciar els estàndards de codificació més habituals.
- 4.** Capacitar-vos per a seleccionar el format de codificació més adequat en funció de la qualitat esperada, el tipus de material i la destinació de la reproducció.

1. Conceptes de codificació

La digitalització dels sistemes d'informació començada als anys vuitanta del segle XX ha comportat la generació constant, ininterrompuda i, sobretot, creixent de dades que s'emmagatzemen en dispositius o es transmeten per sistemes de comunicacions que no han evolucionat al mateix ritme quant a capacitat. La situació s'accelera amb l'auge dels productes i serveis multimèdia, vertaders consumidors incansables de tots els dispositius d'emmagatzematge i culpables directes de l'"estrès" sobre la qualitat de serveis dels proveïdors de xarxes de dades. No hi ha mai un disc prou gran, no hi ha mai una xarxa prou ràpida...

Deixant de banda l'exageració, la **necessitat de comprimir la informació** és lògica: les imatges requereixen força més espai en bits que el text. L'àudio i el vídeo fins i tot demanen encara més en termes d'emmagatzematge i velocitat de transmissió. Hi ha unes quantes xifres que il·lustren aquest aspecte i introdueixen així alguns conceptes bàsics que farem servir al llarg de l'assignatura:

- a) Una pàgina d'aquest llibre conté unes 50 línies de 80 caràcters; si cada caràcter es pot representar, emmagatzemar i transmetre digitalment amb 1 byte (8 bits), cada pàgina ocupa 32.000 bits.
- b) Un disseny d'un edifici en vectors (és a dir, creat amb línies, com els dissenys d'AutoCAD o una font escalable) conté uns 500 traços; si cada traç es representa només per les coordenades d'inici i final (quatre nombres en total) i cada nombre per 2 bytes, la imatge ocupa 32.000 bits.
- c) La música en qualitat CD representa 44.100 mostres de so cada segon; si les mostres són de 16 bits i és estèreo, es generen 1.411.200 bits per segon (gairebé 1,5 megabits per segon o, dit d'una altra manera, 1,5 Mbps).
- d) Una imatge fotogràfica de 720 per 576 píxels conté 414.720 píxels en total; si la imatge és en blanc i negre, cada píxel es representa per 1 byte, de manera que ocupa 3.317.760 bits (més de 3 Mb).
- e) Si la imatge anterior és en color, cada píxel es representa per 3 bytes (un per al vermell, un altre per al verd i un altre per al blau), de manera que ocupa el triple, és a dir, 9.953.280 bits (gairebé 10 Mb).
- f) El so digital envoltant pot arribar a representar gairebé 9 Mbps sense tractament.

g) En el cas de vídeo, un clip de qualitat estàndard de 625 línies a 25 quadres per segon necessita, sense comprimir-lo, uns 216 Mbps.

h) Si fos HD estàndard de 1.080 línies (1080i), augmentaria a 1,15 Gbps; o fins i tot el doble, 2,30 Gbps, si fos HD pur (1080p).

i) Avui dia les solucions propietàries de cinema digital, Ultra HD i Hi-Vision, estan a l'entorn de les 4.000 línies de resolució, i no es podrien transmetre sense compressió per sota dels 10 Gbps.

j) Des del 2005 s'han fet emissions experimentals en Super Hi-Vision (8.000 línies), per exemple en els Jocs Olímpics del 2012. Avui dia el senyal pur de 24 Gbps no es podria transmetre de cap manera, ni tan sols en fibra òptica, si abans no es comprimís, i encara menys per Internet.



Els formats de 4.000 línies s'anomenen 4K, i els de 8.000, 8K. Actualment només es fan servir en postproducció de vídeo HD i cinema.

Il·lustració 1. Televisor LED Super Hi-Vision de 8.000 línies presentat en l'IBC 2012 a Amsterdam

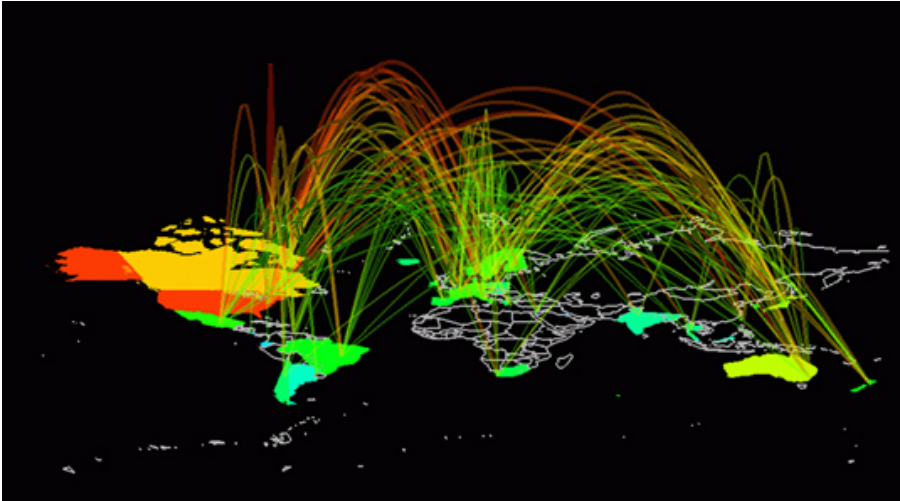


Foto de l'autor

Els requisits de memòria i velocitat de transmissió per a aquestes informacions s'han de reduir de manera notable perquè els puguin gestionar els equips actuals, amb capacitats d'emmagatzematge a l'entorn de gigabytes i memòries intermèdies¹ a l'entorn de megabytes, i de sistemes de transmissió a l'entorn d'uns quants megabytes per segon. Ens cal comprimir, i en alguns casos de manera dràstica, encara que comporti la pèrdua de part del contingut.

⁽¹⁾En anglès, *buffers*.

Il·lustració 2. Representació virtual dels grans fluxos de dades entre els principals països



L'impacte de la compressió

A banda del gigabyte (GB) i del terabyte (TB) hi ha unitats que cada any ens són més familiars: el petabyte (PB, 1.024 GB), l'exabyte (EB, 1.024 PB), el zettabyte (ZB, 1.024 EB) i el yottabyte (YB, 1.024 ZB). Per exemple, el trànsit global d'Internet cada segon és a l'entorn de 20 TB, gairebé el 70% dels quals són transmissió en temps real (*streaming*) de dades de vídeo, el 10% transferència de fitxers, el 15% correus electrònics i navegació per webs, etc.

Així, doncs, la distribució de continguts (sia multimèdia o d'un altre tipus) pot representar prop del 80% del trànsit de la Xarxa en qualsevol moment, i sens dubte comprimits. Per això, qualsevol tècnica que pugui comportar una millora de la compressió tindrà un impacte econòmic de grans magnituds.

1.1. Codificació i compressió

Els termes *codificació* i *compressió* se solen fer servir de manera equivalent, però això és incorrecte: *codificació* és senzillament expressar una informació d'una manera diferent utilitzant un nou codi, mentre que *compressió* és reduir-ne la mida total en bits. La compressió és un cas particular de codificació en el qual el nou codi utilitza menys elements per a representar la mateixa informació.

Per exemple, la taula ASCII codifica els caràcters expressant-los en format binari, de manera que en el procés no hi ha cap compressió. En canvi, una codificació que empra una eina informàtica que representi un fitxer de dades amb un altre de mida més petita en bits (com .zip o .rar a Windows, .sitn a Mac OS X o .tgen a Linux) és un cas de compressió.

La compressió ha d'estar ben especificada tant en el codificador d'origen com en el decodificador del destinatari, perquè així s'assegura que la recepció és correcta. En l'auge d'Internet a la fi dels anys noranta del segle passat van aparèixer molts formats "propietaris" (és a dir, exclusius d'una empresa) que obligaven a l'ús de programes propis per a visualitzar-los, incompatibles amb els de la competència: per a RealVideo i RealAudio de RealNetworks s'havia de fer servir l'aplicació RealPlayer, o el reproductor de Windows si el fitxer tenia

La taula ASCII

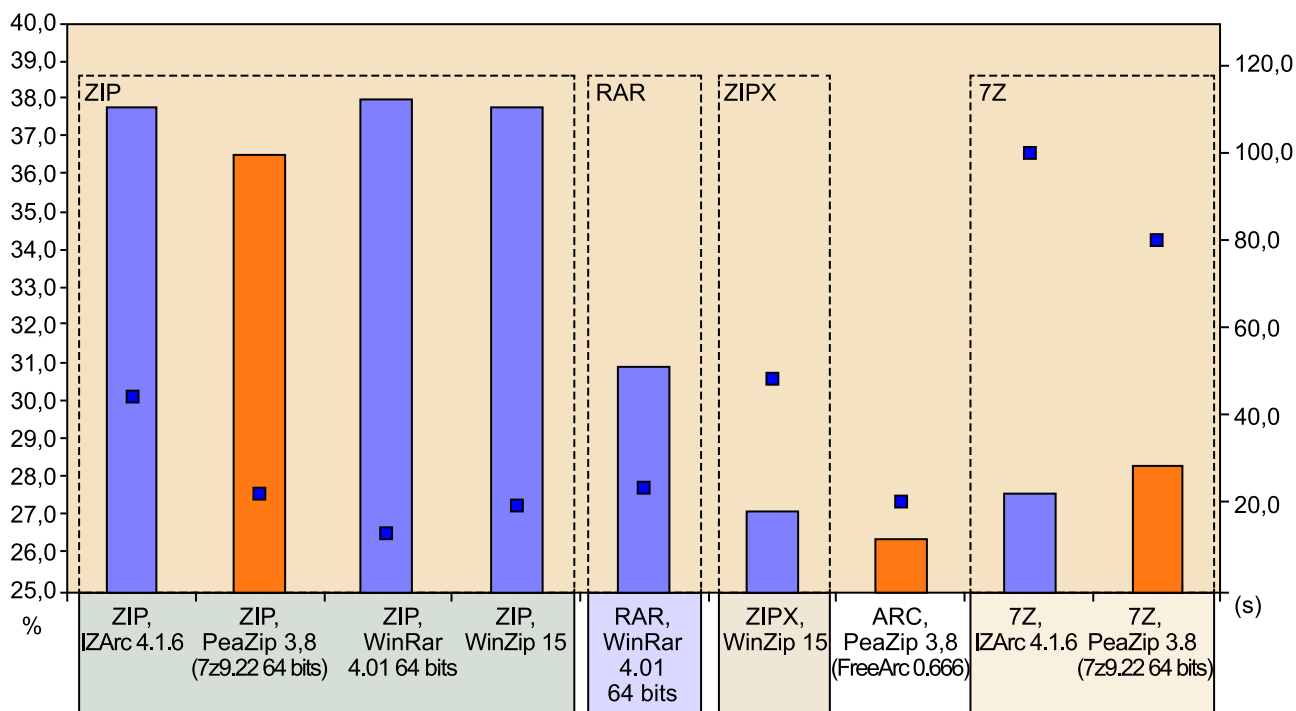
Els ordinadors i tots els sistemes digitals en general només interpreten nombres. La taula ASCII és una convenció que assigna un nombre a cada caràcter per a poder-lo representar en sistemes digitals.

el format Windows Media Video de Microsoft. Fins ben entrat el segle XXI no van aparèixer aplicacions capaces de funcionar amb diferents formats, cosa que va minimitzar el caos i va facilitar la universalització d'aquests formats.

En l'especificació d'una **compressió**, o **còdec**, es representa el vídeo original mitjançant una codificació, és a dir, mitjançant una gran sèrie de valors fàcils de manejar. Idealment, aquesta codificació ha de ser eficient (ha d'utilitzar la mínima quantitat d'informació possible, és a dir, bits) i eficaç (ha de representar de la manera més fidel que es pugui el vídeo original).

És clar que aquests dos objectius estan en conflicte i cal un compromís entre l'un i l'altre. En general un estàndard de còdec presenta diversos **perfils** o configuracions amb diferents valors de compromís entre eficàcia i eficiència amb la finalitat d'adaptar-se a una quantitat més gran de sistemes de vídeo digital.

Il·lustració 3. Raó de compressió (barres) i temps de càlcul (punts)



En ZIP, l'eficiència de la compressió varia segons l'aplicació que es fa servir i també el temps que es requereix per a fer-la. El resultat, però, sempre és un fitxer del qual es poden recuperar totes les dades sense perdre'n cap, és a dir, que és eficaç

Encara podem perfilar més la definició de *compressió*: quan diem que ha de representar la mateixa informació, no volem dir que la informació final hagi de ser exactament la mateixa que l'original en tots els casos, és a dir, que la compressió pot ser sense pèrdues² o amb pèrdues³. En el primer cas, podem fer el procés invers per tornar a tenir la mateixa informació original exacta sense perdre'n gens ni mica; per exemple, tots els fitxers .zip o .rar permeten recuperar el fitxer original. En el segon cas, acceptem una degradació del missatge a fi d'aconseguir una compressió efectiva, sempre que no faci inservible la in-

⁽²⁾En anglès, *lossless*.

⁽³⁾En anglès, *lossy*.

formació que porta; per exemple, en copiar un fitxer d'àudio del CD original a l'iTunes es codifica automàticament en un format d'àudio amb pèrdues (AAC, MP3, etc.) molt més reduït de mida però amb una experiència sonora molt semblant a l'original, encara que no exacta.

1.2. Redundància i entropia

A què ens referim quan parlem d'*informació inservible*? En molts casos la informació que es transmet té **dades redundants**, informació previsible o repetitiva que es pot comprimir de manera intensa, com per exemple uns quants espais en blanc consecutius en un text, o uns quants píxels consecutius d'un mateix color en un fax. La reconstrucció d'aquesta informació en descomprimir és exacta.

En la **compressió sense pèrdues** s'elimina la redundància de la informació, sense tocar gens ni mica la informació fonamental.

L'entropia és una mesura de la quantitat d'informació mitjana que tenen els símbols que es fan servir per a representar aquesta informació. Si podem representar una imatge amb un conjunt de símbols petit, cadascun amb una gran quantitat d'informació, la compressió serà molt alta i no tindrà cap pèrdua. Així, una informació poc probable té més informació que una de més probable, de manera que es poden assignar més bits a la poc probable i menys a la més probable. Algunes tècniques de codificació que es basen en aquest principi entròpic són la base de la compressió sense pèrdues de fitxers de dades i també alguna de les etapes dels processos de compressió amb pèrdues.

Avui dia les compressions sense pèrdues poden arribar a relacions de compressió 2:1, la qual cosa és un èxit, però la major part de vegades no n'hi ha prou amb això, de manera que la compressió amb pèrdues és la tècnica que impera en el món de la imatge i el so, en el qual es necessiten taxes de reducció molt més altes a costa d'una pèrdua controlada de fidelitat.

La base de la **compressió amb pèrdues** és l'eliminació de les dades irrellevants en funció de les limitacions del destinatari, com per exemple les petites variacions de color entre píxels pròxims (ja que l'ull humà té poca resolució per al color) o els sons de freqüències superiors als 16.000 Hz (perquè no hi ha gaires éssers humans adults que siguin capaços de sentir-los). En la **compressió amb pèrdues**, la informació irrellevant se suprimeix del tot, cosa que modifica la informació de manera permanent, encara que tolerable, per al destinatari a qui vagi adreçada.

Compressió sense pèrdues

La compressió sense pèrdues és la ideal per a fitxers de dades (.zip, .rar, .sit, .tg, etc.), alguns formats d'imatge com GIF, PIC, un mode de treball específic del format de compressió d'imatge JPEG 2000, etc.

L'entropia en la teoria de la informació

En l'àmbit de la teoria de la informació, l'entropia és un concepte una mica diferent del que representa en la termodinàmica o la mecànica.

Compressió amb pèrdues

Vegem alguns exemples de formats estàndard de compressió amb pèrdues:

- Formats de compressió d'imatge com JPEG, EZW, SPIHT i gairebé tots els modes de treball de JPEG 2000.
- Formats de compressió d'àudio com AAC o MP3.
- Formats de compressió de vídeo com la família MPEG (1, 2, 4).

És clar que l'èxit de qualsevol compressió depèn de la quantitat d'informació redundant o irrellevant que hi hagi en el missatge, de manera que el rendiment de la compressió sol variar gairebé constantment. L'entropia és una mesura de la informació essencial d'un missatge (la diferència entre la quantitat total de dades que conté el missatge i la redundància d'aquestes dades). Com més entropia hi hagi, més dades hi haurà que no es podran eliminar, per la qual cosa la codificació amb pèrdues no rendirà gaire. Com menys entropia hi hagi, molt més efectiva serà la codificació amb pèrdues.

1.3. Mesura de la compressió d'imatge

Per a mesurar la compressió assolida en una imatge i així poder comparar diferents sistemes de compressió, com farem en l'assignatura, fem servir el **factor de compressió**, un valor numèric que augmenta amb la compressió assolida.

Donada una imatge original i l'equivalent comprimit d'aquesta imatge, el **factor de compressió** és:

$$\text{Factor de compressió} = \frac{\text{Pes de la imatge "original"}}{\text{Pes de la imatge "comprimida"}}$$

Hem de pensar d'indicar els dos pesos sempre en la mateixa unitat (bits, bytes, kilobytes, etc.).

Una altra manera de representar la compressió és indicant el nombre de bits que necessita cada píxel de la imatge, el **bit per píxel (bpp)**:

$$\text{bpp} = \frac{\text{Pes de la imatge (expressat en bits)}}{\text{Nombre total de píxels de la imatge}}$$

De manera senzilla podem deduir que el bpp d'una imatge original i el bpp d'aquesta imatge comprimida estan relacionats pel factor de compressió entre l'una i l'altra mitjançant l'expressió següent:

$$\text{Factor de compressió} = \frac{\text{bpp de la imatge "original"}}{\text{bpp de la imatge "comprimida"}}$$

Exercici

Si una imatge és de 1.600×1.200 píxels amb 24 bits de color per píxel i es comprimeix en un format tal que el fitxer resultant és de 230 KB, quin és el bpp de la imatge original?

I el de la comprimida? Quin és el factor de compressió aconseguit? Paeu atenció a les unitats i no barregeu bits amb bytes.

1.4. Mesura objectiva i subjectiva de la qualitat d'imatge

Avui dia es continuen investigant tècniques quantitatives que puguin assignar un valor numèric a la qualitat visual de qualsevol imatge, que ens proporcioni una idea de fins a quin punt ha quedat degradada després d'aplicar-hi certa compressió amb pèrdues i ens ajudi així a triar l'estàndard de compressió més adequat.

En tot cas, presentem una mesura, l'error quadràtic mitjà, una manera de calcular les diferències que hi ha entre una imatge i aquesta mateixa imatge comprimida. Bàsicament una aproximació per a diferenciar fins a quin punt són diferents, cosa que l'ull humà fa de manera natural a l'instant però que matemàticament és molt difícil de concretar, ja que no se saben amb detall els mecanismes del cervell humà que fan possible aquesta gesta. La idea de l'error quadràtic mitjà consisteix a fer una versió monocroma de la imatge (els colors no es tenen en compte), restar píxel a píxel l'una de l'altra i sumar finalment les diferències al quadrat (així totes són positives i no pot passar que n'hi hagi algunes que es cancel·lin amb les altres). El resultat és un valor que com més s'acosta a zero, més s'assemblen les imatges.

Si la imatge original O és de mida $N \times M$ píxels, i la imatge C és la mateixa però comprimida, l'error quadràtic mitjà ε^2 resulta:

$$\varepsilon^2 = \frac{1}{N \cdot M} \sum_{i=1}^N \sum_{j=1}^M (o[i, j] - c[i, j])^2$$

en què $o[i, j]$ és el valor del píxel de la fila i columna j de la imatge O monocroma, i $c[i, j]$, el de la imatge C monocroma.

Després de veure el format JPEG veurem una manera molt més visual i entretinguda de detectar els errors mitjançant l'aplicació Photoshop, que és capaç de calcular la imatge diferència, és a dir, una resta visual píxel a píxel entre, per exemple, una imatge sense comprimir i l'equivalent comprimit d'aquesta imatge, que posa en evidència la informació que es perd en la compressió.

Una altra mesura derivada d'aquesta és la **relació pic de senyal/soroll** ($PSNR^4$):

$$PSNR = 10 \cdot \log_{10} \left(\frac{MAX^2}{\varepsilon^2} \right)$$

⁽⁴⁾PSNR és la sigla de l'expressió anglesa *peak signal-to-noise ratio*.

en què *MAX* és el valor màxim que pot prendre un píxel, en general 255. El resultat és en decibels (dB), i els valors habituals se solen situar entre 35 dB i 50 dB. Com més gran és el valor, més s'assembla a l'original la imatge comprimida.

La relació pic de senyal/soroll, però, és una mesura que no és del tot fiable, ja que els errors visuals són més percebuts per l'ull humà en funció de l'entorn en què apareixen, cosa que no es valora amb aquestes eines, més globals.

La segona família de mesures es basa en enquestes que recullen la impressió subjectiva dels usuaris davant les dues imatges, i valoren la comparativa mitjançant uns pesos. Per exemple:

| Rànquing | Descripció de la comparativa |
|----------|------------------------------|
| 9-10 | Molt bona |
| 7-8 | Bona |
| 4-6 | Acceptable |
| 2-3 | Dolenta |
| 0-1 | Molt dolenta |

1.5. Classificació de les tècniques de compressió

La principal classificació de les tècniques de compressió divideix els còdecs en dues grans famílies, tal com hem indicat breument al principi de l'apartat: sense pèrdues o amb pèrdues.

1.5.1. Compressió sense pèrdues

Les compressions sense pèrdues (*lossless*) proporcionen la màxima qualitat visual, i es fan servir en entorns de documentació o postproducció d'imatge i vídeo en els quals la imatge s'ha de mantenir a la màxima qualitat d'origen.

Els còdecs Matrox, AJA o Blackmagic es van popularitzar en aquest entorn per a vídeo de definició estàndard, i amb l'arribada de l'alta definició es continuen utilitzant en la versió HD. El format DPX de Kodak, basat en el seu clàssic sistema de digitalització Cineon (1993), permet treballar en HD amb vídeo en RGB pur i mostratge 4:4:4 a 10 bits per mostra. Les millores del format de fitxer d'imatge OpenEXR d'Industrial Light & Magic (2007) com un format obert



sense llicència amb una variada col·lecció d'algorismes de compressió sense pèrdua capaços de comprimir fins i tot en ràtios 2:1 ha modificat el panorama de la postproducció.

No obstant això, per a reduir les necessitats d'espai i d'amplada de banda de treball en HD (entre 1 i 1,5 Gbps), s'han començat a aplicar discretament còdecs de compressió amb pèrdues per a reduir així els requisits d'emmagatzematge en *arrays* de discos.



En resum, les tècniques sense pèrdua queden delimitades per a entorns específics i dedicats, i sempre tenen taxes de compressió molt baixes (per no dir que gairebé inexistents si s'apliquen sobre un material de vídeo visualment ric i molt canviant).

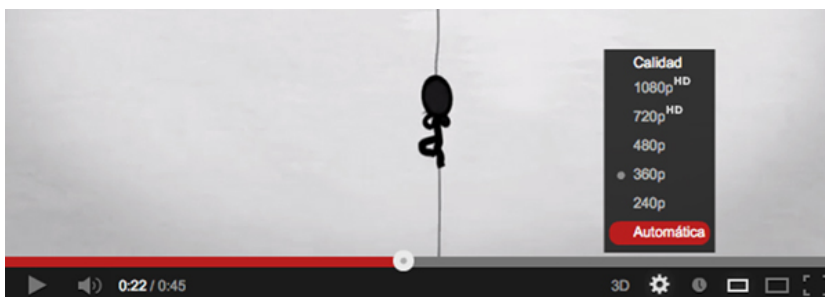
1.5.2. Compressió amb pèrdues

La **compressió amb pèrdues** (*lossy*) no implica renunciar a una qualitat final de treball, sinó que defineix la qualitat adequada per al tipus de destinatari a qui va adreçat.

Qualitat segons el destinatari

Un clip informatiu penjat a YouTube no cal que sigui de màxima qualitat si l'objectiu és informar; en canvi, un curtmetratge penjat també a aquesta plataforma pot haver de ser HD per a presentar la riquesa cromàtica dels seus escenaris. En tots dos casos, al servidor de YouTube hi tenim el fitxer de vídeo amb certa qualitat inicial, i YouTube el redueix visualment segons la definició que requereixin els possibles destinataris, sempre en sentit descendent, de la qualitat original a qualitats inferiors:

Il·lustració 4. Opcions de visualització d'un vídeo 1080p a YouTube



En el cas de reproducció en temps real (*live streaming*), la compressió amb pèrdues és l'aposta necessària i imprescindible per a facilitar l'arribada de la imatge i el so amb la cadència necessària perquè s'entengui. No hi ha prou temps per a fer una codificació òptima com es faria en un vídeo emmagatzemat: "obligacions del directe". I així i tot no assegura l'èxit en aquest objectiu perquè hi ha altres elements que ho poden impedir, com veurem més endavant.

Il·lustració 5. Canal de notícies 324

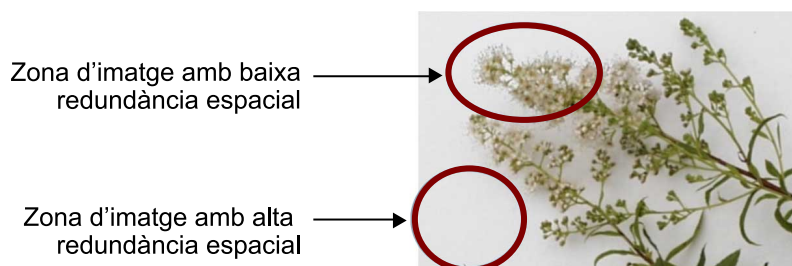


Hi ha molts canals de televisió que emeten en temps real per Internet

Les tècniques més comunes de compressió es basen, d'una banda, en l'eliminació de la redundància espacial i temporal que hi ha en el vídeo "cru" –característiques habituals del vídeo– i, de l'altra, en l'aprofitament de les limitacions dels sentits humans:

a) **Redundància espacial:** es basa en la suposició que els píxels d'una zona d'una imatge són molt semblats als de les zones que l'envolten, excepte on hi hagi un perfil (la transició d'un objecte a un altre representat en la imatge). L'estàndard JPEG d'imatge es basa en aquestes similituds per a comprimir la imatge fixa, de manera que només emmagatzema les diferències lleus entre grups de píxels i els seus veïns. Veurem que l'estàndard MPEG també ho aplicarà, en aquest cas a vídeo, sobre alguns dels seus quadres d'imatge.

Il·lustració 6. Una imatge amb zones molt diferenciades de detall



En altres paraules, les zones homogènies de la imatge són candidates a una bona compressió espacial amb poc esforç. En aquest punt, però, el que hi va és el detall, és a dir, la riquesa de la imatge. El **compromís entre compressió i qualitat d'imatge** diferenciarà un bon plantejament de compressió d'altres no tan encertats.

b) **Redundància temporal:** es basa en la suposició que no hi ha gaires diferències entre un fotograma i els altres fotogrames que hi ha immediatament abans o després.

Il·lustració 7



Un *pan* o *escombratge suau* de la càmera d'esquerra a dreta sobre un aparador presenta molt poques diferències entre dos fotogrames consecutius del vídeo generat. Sense haver de calcular la imatge diferència es pot assegurar que hi ha una alta redundància temporal

c) **Redundància psicovisual:** l'ull humà té una resposta limitada als detalls espacials fins (poca sensibilitat a l'alta freqüència espacial), i és menys sensible al detall (llegiu-hi "als errors") quan aquest detall és prop de les vores d'objectes o en un canvi de pla. En conseqüència, aquestes limitacions poden camuflar efectes en la imatge descodificada quan es defineix l'algorisme de compressió amb pèrdues.

En els estàndards de compressió d'imatge fixa, àudio i vídeo que presentarem en els mòduls següents, veurem l'ús separat o combinat d'aquestes tres tècniques.

1.6. Tractament del color: representació i submostratge

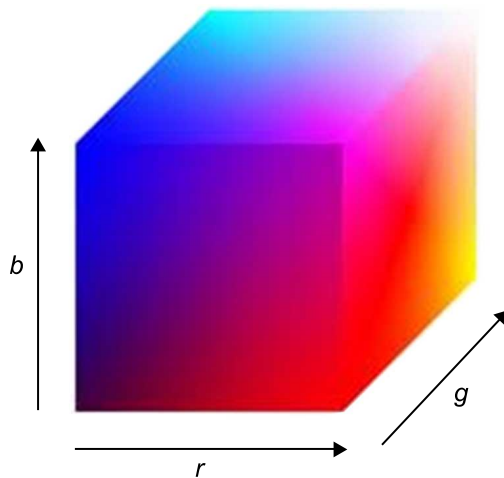
Com que avui dia qualsevol imatge i vídeo es genera en color, aquest color té un tractament específic molt habitual en tots els estàndards de compressió que presentarem tot seguit. En primer lloc, el color es pot representar de diferents maneres o en diferents espais, uns més adequats que altres a l'hora de codificar. En segon lloc, se solen reduir el nombre de mostres de color (submostratge) en tots plegats.

1.6.1. Representacions del color: espais RGB i YCbCr

Qualsevol píxel d'una imatge es pot trobar mitjançant les seves dues coordenades X i Y , però a més té tres valors enters que en representen el color en un espai de color. Els espais de color més coneguts són RGB, YCbCr (per a vídeo digital) i YUV (per a vídeo analògic).

RGB es basa en els colors primaris: vermell (R), verd (G) i blau (B). És un espai de color intuïtiu en el qual la majoria de colors es poden representar com una combinació d'aquests tres colors.

Il·lustració 8. Colors RGB



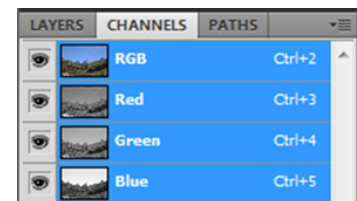
Atès que la gamma és infinita en qualsevol d'aquests colors, se sol reduir a 256 valors enters per color, del 0 al 255. Així, $R = 0$ és el vermell negre, i $R = 255$, el vermell més vívid i pur. Cal especificar, però, que aquesta codificació es fa servir per pura conveniència.

Els altres dos espais de color en què també es pot definir qualsevol color amb tres valors són equivalents a RGB, però plantegen cada color com una combinació de blanc i negre i dos components de color.

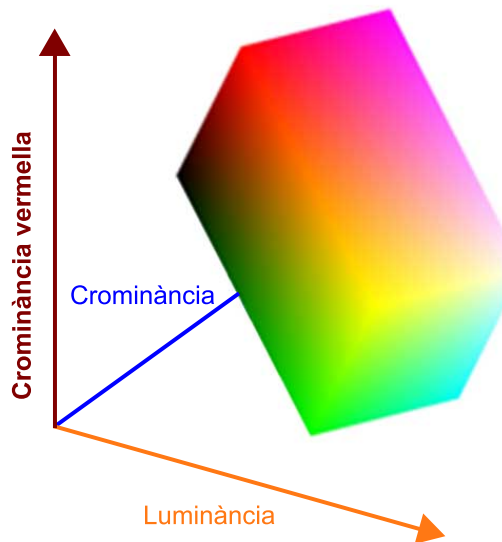
Així, a YCbCr el valor Y és el valor de luminància (brillantor), Cb el valor de crominància blava i Cr el de crominància vermella. El valor de crominància verda es pot deduir dels altres tres. L'espai YUV també segueix la mateixa filosofia.

Crominància blava

S'anomenen *crominàncies blava i vermella* però no són els components de color vermell i blau, ni tan sols corresponen a colors, sinó que només són valors numèrics que fins i tot poden ser negatius. Aquests senyals també s'anomenen *diferències de color*.



Il·lustració 9. Canals de color



Per aritmètica es poden deduir els valors Y , Cb i Cr d'un senyal digital (per a 8 bits per mostra de color) a partir dels valors RGB d'un píxel. Suposant que el rang de valors R , G o B és $[0 \dots 255]$, tenim:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 0,299 & 0,587 & 0,114 \\ -0,169 & -0,331 & 0,500 \\ 0,500 & -0,419 & -0,081 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Els valors Y , Cb i Cr

Aquestes fórmules només són vàlides per als formats que possibilitin emmagatzemar tot el rang de valors de cada component, com per exemple: JPEG.

en què el rang de valors possibles de Y , Cb i Cr és $[0 \dots 255]$. L'operació contrària és:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1,4 \\ 1 & -0,343 & -0,711 \\ 1 & 1,765 & 0 \end{bmatrix} \cdot \begin{bmatrix} Y \\ Cb - 128 \\ Cr - 128 \end{bmatrix}$$

La raó per la qual se sol utilitzar més l'espai YCbCr que l'espai RGB és doble: d'una banda, descomponem un color en un component que representa la luminància i en dos de color o crominància, i així dona peu a poder-los processar separatament; de l'altra, l'estructura de bastons i cons de l'ull segueix el mateix principi i té menys sensibilitat al color que a la brillantor o luminància. La comprovació és senzilla: si posem un fil de color vermell a cinc metres d'un observador, en general detectarà el fil, però no el color. El color, pel que sembla, no era gaire rellevant per a la supervivència dels nostres ancestres, o com a mínim no ho era tant com la brillantor de llum.

Sabent, llavors, que cada quadre d'imatge pot ser representat pels seus tres quadres equivalents Y , Cb i Cr , i sabent que la informació visual és sobretot al quadre Y , es podrà comprimir el quadre Y sense gaires pèrdues (és a dir, una compressió més aviat suau) i es podran comprimir amb més pèrdues (més compressió) els quadres Cb i Cr .

Ocorre així en l'estàndard **JPEG de compressió d'imatge fixa**. La primera etapa del procés d'aquest estàndard és convertir la imatge de l'espai de color RGB (tal com es genera en el sensor de càmera) a l'espai YCbCr.

1.6.2. Submostratge de color

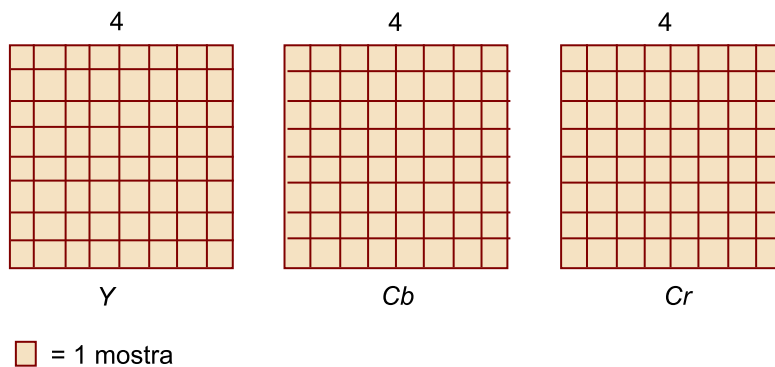
Com que l'ull humà és menys sensible al color, no és imprescindible mantenir totes les mostres dels components *Cb* i *Cr* de tots els píxels, sinó solament les d'uns quants. Aquesta idea s'aplica en el mer procés de mostratge del color en cada imatge i s'anomena **submostratge**.

Així, es fa un mostratge més dens en l'equivalent *Y* i no tan dens en els equivalents de color sense afectar seriosament la qualitat final.

La sistemàtica de mostratge està estandarditzada en la família MPEG, basada en YCbCr:

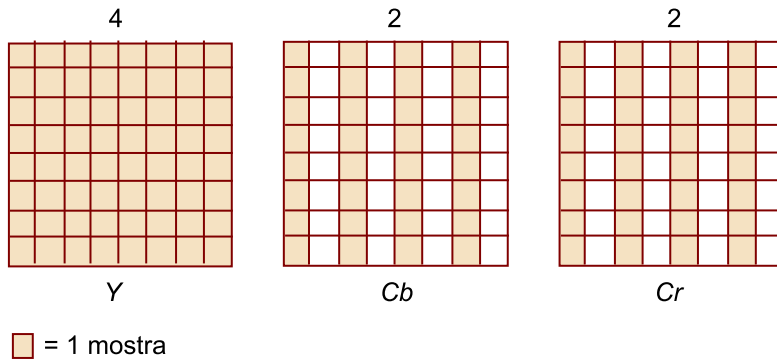
a) **4:4:4**: els tres components (*Y*, *Cb* i *Cr*) presenten la mateixa resolució i hi ha una mostra de cada component en cada píxel. Només és per a entorns professionals en què no es vol compressió (HDCAM, MPEG-2, H.264):

Il·lustració 10. Mostres de cada component en el cas 4:4:4



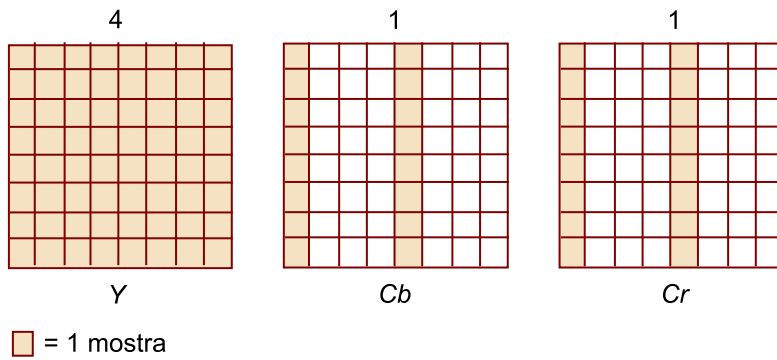
b) **4:2:2** (conegut també com a *YUY2*): presenta dos components de crominància per cada quatre de luminància (Digital Betacam, DVCPRO 50, MPEG-2, H.264):

Il·lustració 11. Mostres de cada component en el cas 4:2:2



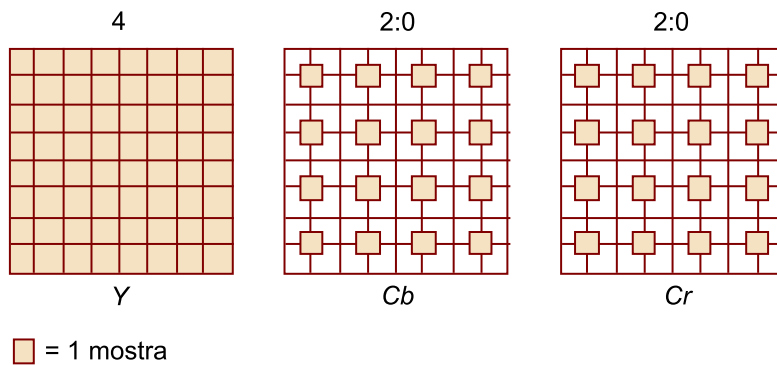
c) **4:1:1**: presenta un component de crominància per cada quatre de luminància:

Il·lustració 12. Mostres de cada component en el cas 4:1:1

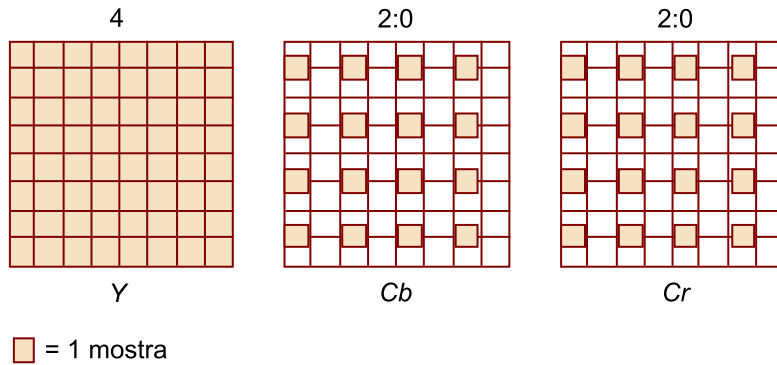


d) **4:2:0** (conegut també com a *YV12*): presenta un component de crominància per cada quatre de luminància (DVD, MPEG-1, MPEG-2, MPEG-4), encara que segons l'estàndard se selecciona un lloc diferent del quadre per a realitzar la mostra dels components de color:

Il·lustració 13. Mostres de cada component en el cas MPEG-1 (4:2:0)



Il·lustració 14. Y en el cas MPEG-2 (4:2:0)



En resum, els tres patrons tenen una distribució dels píxels que es mostregen en cada cas especificadament:

- En el cas 4:4:4, es mostregen tots i cadascun dels píxels dels tres components de cada quadre.
- En el cas 4:2:2, es mostregen tots els píxels del component de luminaància Y i la meitat de cada component de color.
- En el cas 4:2:0, es mostregen tots els píxels del component de luminaància Y , i només un de cada quatre en els components de color, que varien segons la norma.

En tots aquests casos, cada mostra aconseguida es representa després amb 8 o 10 bits, segons la precisió de la mostra que es vulgui.

1.7. Exercicis

Exercici 1

Busqueu a Internet estimacions de les mides dels conceptes següents:

- Quants petabytes ocupa Facebook?
- Quants petabytes diaris gestiona Google?
- En quants exabytes s'estima la mida total d'Internet?

Exercici 2

Calculeu els valors Y , Cb i Cr d'un color que té components RGB de valors $R = 60$, $G = 190$ i $B = 210$. Feu l'operació inversa per confirmar que els valors obtinguts són correctes.

Exercici 3

Donada una imatge en color de $N \times M$ píxels en la qual cada component R , G i B és un valor emmagatzemat en 8 bits (és a dir, 24 bits per píxel):

- a) Quin pes total en bytes té la imatge?
- b) Es transforma a YCbCr mitjançant un mostreig 4:4:4. Quin pes tindrà ara?
¿Quin factor de compressió s'ha assolit?
- c) Si la transformació hagués estat 4:2:2, quin pes tindria?, quin factor de compressió s'assoliria?
- d) Feu la mateixa operació si fos 4:2:0.

2. Codificació d'imatge fixa

Les imatges requereixen força més espai que el text, i l'àudio i el vídeo encara en demanen més en termes d'emmagatzematge i velocitat de transmissió.

Començarem veient el primer sistema de codificació d'imatge fixa i introduïrem una sèrie de conceptes que farem servir al llarg de tot el curs.

2.1. El format JPEG

JPEG és un comitè (Joint Photographic Experts Group), no un estàndard de codificació, que a la fi dels anys vuitanta va idear un procés pel qual una imatge digital es podia comprimir mantenint un compromís amb la qualitat visual.

En aquells anys ja començava a ser necessari un sistema perquè les imatges no ocupessin els reduïts espais d'emmagatzematge digital que hi havia llavors i es poguessin transmetre per línies de telefonia digital de baixa velocitat. En general, el procés és prou robust i de resultats acceptables, sobretot amb fotografies naturals i imatges reals, i s'arriba a una mitjana de compressió de 10:1 amb pèrdues imperceptibles visualment, de manera que continua essent un format que es fa servir molt.

Per primera vegada en aquesta assignatura hem de distingir entre el sistema de codificació (còdec) de la imatge, que és JPEG, i el **contenidor** o format de fitxer on es guarden les dades resultants de la codificació, del qual n'hi ha dos tipus, JFIF i EXIF. Així, JPEG/JFIF és el format clàssic, mentre que JPEG/EXIF possibilita afegir metadades (dades addicionals associades a la imatge) com, per exemple, la data i hora de la captura, la velocitat de l'obturador, el nom de la càmera o les dades de configuració. Avui dia hi ha alguns telèfons mòbils amb càmera i GPS integrats que afegeixen la longitud i latitud en què es va fer la captura. El nostre interès se centra en el còdec.

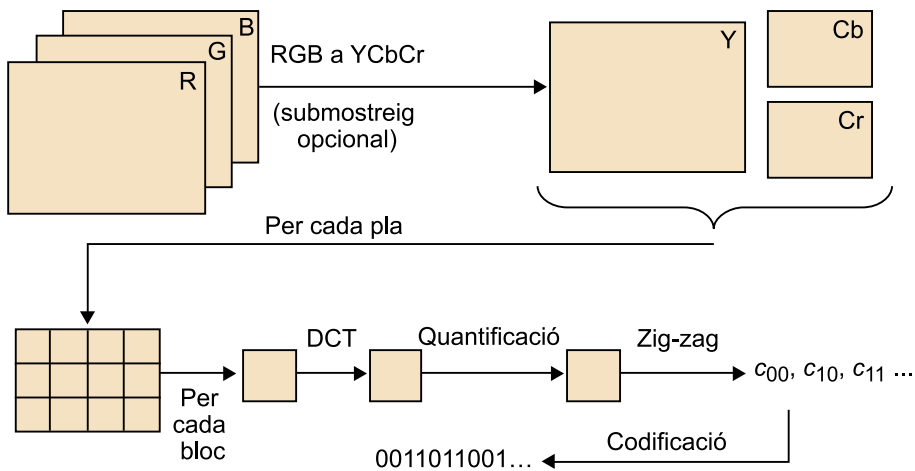
JPEG pot comprimir amb pèrdues o sense pèrdues en funció del mode de selecció de treball, però on destaca és en els modes amb pèrdues, ja que aconsegueix compressions de fins a 20-25 sense efectes visuals aparents.

Els tres modes amb pèrdues són el **mode seqüencial** –que és el més utilitzat i amb el qual introduïrem diversos conceptes de codificació rellevants en cada pas del procés de codificació–, el **mode progressiu** i el **mode jeràrquic**.

2.1.1. Mode seqüencial

Un esquema general dels passos que se segueixen en el mode seqüencial és el següent:

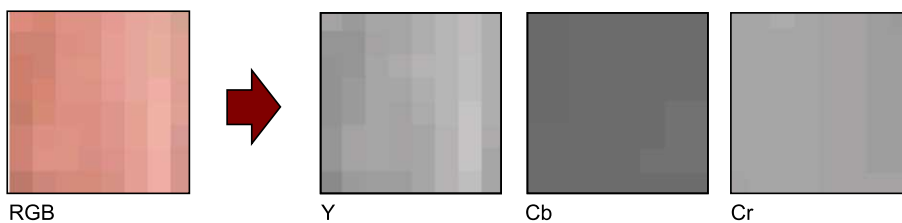
Il·lustració 15



Pas 1. Canvi d'espai de color i submostratge del color

Si la imatge és RGB, primer de tot es transforma a YCbCr. Com hem vist al principi d'aquest mòdul, com que l'ull humà té menys resolució de color, les imatges *Cb* i *Cr* se submostregen i redueixen el nombre de píxels (en proporció 4:4:4, 4:2:2, 4:1:1 o 4:2:0 en funció de la qualitat triada). La imatge *Y* (blanc i negre) es manté inalterada.

Il·lustració 16



Pas 2. Divisió en blocs i seqüenciació

Cadascun dels tres components es comprimeix per separat i es divideix en blocs de 8 × 8 píxels. Així, cada component es converteix en una seqüència de blocs, començant pel que té a la cantonada superior esquerra i acabant pel de la cantonada inferior dreta. A partir d'aquest punt el procés el fa individualment per cada bloc.

Pas 3. Codificació de cada bloc

Cada bloc d'imatge de 8 × 8 píxels es transforma matemàticament en un altre bloc de 8 × 8 valors (també enters) que ja no representen una petita imatge sinó les freqüències espacials d'aquesta imatge. El bloc, doncs, conté la mateixa informació que tenia però amb altres valors: un exemple pur de codificació. I sense pèrdues fins aquest punt, ja que podríem fer el procés invers i recuperar exactament la petita imatge original.

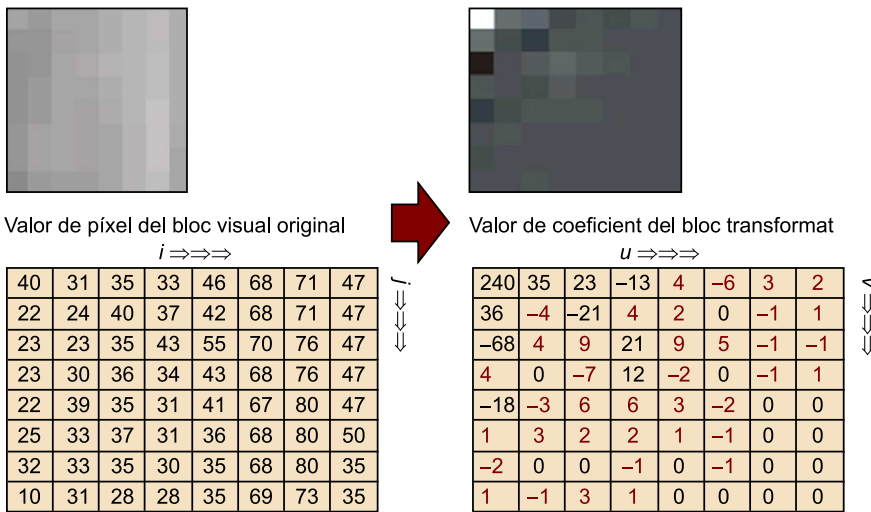
8 × 8 píxels

Els píxels són de valor enter de 0 a 255, però en aquest punt es resta a cadascun 128 perquè estiguin en el marge de -128 a 127. Així, la transformació DCT següent dóna un rang de valors menys extens i més fàcil de codificar.

La transformació utilitzada s'anomena DCT⁵ i és la base de la compressió d'imatge i vídeo avui dia.

⁽⁵⁾DCT és la sigla de l'expressió anglesa *discrete cosine transform* (transformada discreta del cosinus).

Il·lustració 17. Exemple de transformació DCT



Donat el bloc de 8 × 8 píxels de la imatge original, en què el valor del píxel de la columna i fila j és $g_{i,j}$, es pot generar cadascun dels coeficients de la taula transformada, en què $g_{u,v}$ és el coeficient de la freqüència espacial horitzontal u i de la vertical v mitjançant la fórmula de la DCT següent:

$$g_{u,v} = \alpha(u) \cdot \alpha(v) \cdot \sum_{i=0}^7 \sum_{j=0}^7 g_{i,j} \cdot \cos\left[\frac{\pi}{8} \cdot \left(i + \frac{1}{2}\right) \cdot u\right] \cdot \cos\left[\frac{\pi}{8} \cdot \left(j + \frac{1}{2}\right) \cdot v\right]$$

en què $\alpha(u)$ val $1/\sqrt{8}$ si $u = 0$ o $1/\sqrt{4}$ en la resta de casos, i els valors de u i v van de 0 a 7.

Pas 4. Quantificació de cada bloc

Si els dos blocs contenen la mateixa informació, per què bona part dels valors o coeficients del segon bloc són de valor molt reduït, fins i tot nul, respecte als altres? Aquest és el quid del procés: la transformació DCT compacta la informació associada a la imatge en els primers coeficients (situats prop de la cantonada superior esquerra), de manera que ens possibilita eliminar bona part de la resta amb valors nuls o de baix valor (indicats en vermell en la il·lustració 17), i reduir així el nombre total de coeficients que representen el bloc.

Ara bé, quin criteri s’ha de triar per a eliminar els coeficients petits? Què significa *petit*? Per a automatitzar la selecció dels coeficients que s’eliminaran en cada bloc d’una manera raonable es fa servir una taula de referència anomenada **taula de quantificació**, que conté 8×8 valors de pes o ponderació, definits després de moltes proves de qualitat subjectiva fetes amb grups d’observadors. Aquesta taula conté pesos petits en les posicions pròximes a la cantonada superior esquerra i pesos creixents com més lluny són d’aquesta cantonada. Cada coeficient de la imatge transformada es divideix pel pes que li correspon per posició, de manera que els que són lluny de la cantonada superior esquerra queden dividits per pesos elevats, i donen resultat zero (no es tenen en compte els decimals).

| | | | | | | | |
|----|----|----|----|-----|-----|-----|-----|
| 16 | 11 | 10 | 16 | 24 | 40 | 51 | 61 |
| 12 | 12 | 14 | 19 | 26 | 58 | 60 | 55 |
| 14 | 13 | 16 | 24 | 40 | 57 | 69 | 56 |
| 14 | 17 | 22 | 29 | 51 | 87 | 80 | 62 |
| 18 | 22 | 37 | 56 | 68 | 109 | 103 | 77 |
| 24 | 35 | 55 | 64 | 81 | 104 | 113 | 92 |
| 49 | 64 | 78 | 87 | 103 | 121 | 120 | 101 |
| 72 | 92 | 95 | 98 | 112 | 100 | 103 | 99 |

Il·lustració 18. Una taula de quantificació utilitzada en JPEG
Font: Losheller

Si féssim la transformació inversa, anomenada *IDCT*, per recuperar el bloc d’imatge original a partir del bloc transformat i quantificat, l’efecte visual de la pèrdua de coeficients seria mínim. En l’exemple, el fet de descartar els indicats en vermell (il·lustració 17) no afecta sensiblement el resultat “destransformat” (il·lustració 19).

Il·lustració 19. Exemple de transformació DCT inversa



Valors de coeficient del bloc transformat i degudament quantificat

| | | | | | | | |
|----|---|----|-----|---|---|---|---|
| 15 | 3 | 2 | -13 | 0 | 0 | 0 | 0 |
| 3 | 0 | -1 | 0 | 0 | 0 | 0 | 0 |
| -4 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |



Valor de píxel del bloc visual destransformat resultant

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 40 | 30 | 33 | 32 | 42 | 59 | 65 | 44 |
| 22 | 23 | 41 | 33 | 45 | 61 | 69 | 44 |
| 23 | 22 | 36 | 46 | 51 | 73 | 73 | 43 |
| 22 | 31 | 36 | 33 | 41 | 70 | 75 | 44 |
| 23 | 38 | 33 | 31 | 40 | 63 | 81 | 43 |
| 24 | 32 | 34 | 29 | 32 | 63 | 75 | 53 |
| 31 | 29 | 33 | 32 | 32 | 60 | 76 | 35 |
| 8 | 29 | 25 | 25 | 34 | 67 | 75 | 35 |

Sens dubte, el fet d’emmagatzemar en un fitxer els coeficients transformats que ens interessin en comptes de tots els píxels originals donaria una bona taxa de compressió. Si a més descartem els de valors que no es tenen en comp-

te, la taxa millora sensiblement i minimitza així l'efecte visual de les pèrdues introduïdes quan s'inverteix la transformació i es recupera la imatge. A més, la DCT no necessita un processador potent de càlcul; en efecte, es pot executar en maquinari senzill. No obstant això, el procés de compressió JPEG encara no s'ha acabat.

Fem una pausa en el procés d'explicació i raonem els punts següents:

a) Com pot ser que la DCT faci insensible visualment l'eliminació de bona part dels 64 valors?

Després d'anys de proves i comparatives amb totes i cadascuna de les transformacions conegudes, la DCT és la que condensa més bé allò que és important en menys dades sense necessitat de gaire potència de càlcul –factor que als anys vuitanta era crític (el 80386 d'Intel era el processador del moment fins al 1989). Hi ha transformacions més bones, però totes necessiten càlcul intens superior, com, per exemple, JPEG 2000, que veurem més endavant en aquest mateix mòdul.

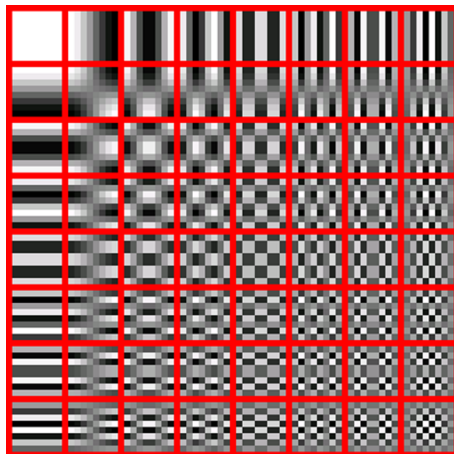
b) Què significa visualment cadascun dels 64 coeficients?

Tal com hem indicat, aquestes dades generades per la DCT representen freqüències espacials, és a dir, cada coeficient, tot sol, generaria un bloc d'imatge simple corresponent a una oscil·lació visual.

Exemple complet de la imatge generada per cada coeficient separatament

La imatge adjunta és un resum del tipus de bloc d'imatge generat per cada coeficient en funció de la posició que ocupa en el bloc transformat. Per exemple, el coeficient que se situa a la cantonada superior esquerra genera tot sol un bloc d'imatge de 8×8 píxels com l'indicat en aquesta mateixa cantonada, un bloc visualment homogeni; el segon coeficient genera un bloc d'imatge de 8×8 píxels semblant a una ona horitzontal de mig període; el tercer, una ona horitzontal d'un període; etc.

Il·lustració 20. Exemple complet de la imatge generada per cada coeficient per separat



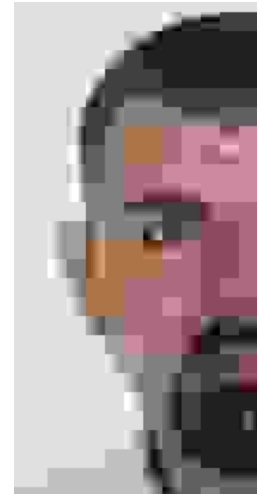
Doncs bé: qualsevol fragment de fotografia en blanc i negre de 8×8 píxels que es triï es pot representar com una suma d'aquests 64 blocs, degudament ponderats (multiplcats) per un valor. Si tinguéssim temps (anys) i paciència, podríem anar provant sumes d'aquests blocs degudament amplificats cadascun per un valor que triaríem nosaltres ma-

teixos. El dia que aconseguíssim la suma perfecta que reproduís exactament el fragment de fotografia, descobriríem que els 64 valors seleccionats coincideixen amb els generats per la senzilla DCT.

El primer coeficient del bloc és el pes o valor de la freqüència zero, que anomenem **component de contínua o baixa freqüència (visual)**, el qual, com es pot veure (il·lustració 21), recrea el valor mitjà de brillantor de tot el bloc i genera per ell mateix un bloc visual de 8×8 homogeni. Així, si esborrèssim tots els components del bloc excepte aquest i apliquéssim la transformada inversa per tornar a tenir un bloc d'imatge, tindríem el que anomenem *efecte mosaic*.

L'ull humà és molt sensible a aquest valor mitjà i s'adona de qualsevol error en aquest valor, de tal manera de qualsevol variació que hi hagi entre dos quadres consecutius genera un efecte mosaic. Per això JPEG cuida com cap altre el valor d'aquest coeficient, com veurem en el pas següent.

Els altres 63 coeficients s'anomenen **coeficients d'alterna**, o **coeficients d'alta freqüència**, i són responsables de les variacions de brillantor a partir d'aquest nivell de base. Els blocs d'imatge amb molt detall visual, com que s'hi aplica la DCT, tenen molts coeficients d'aquest tipus amb gran valor, al contrari dels blocs de poc detall.

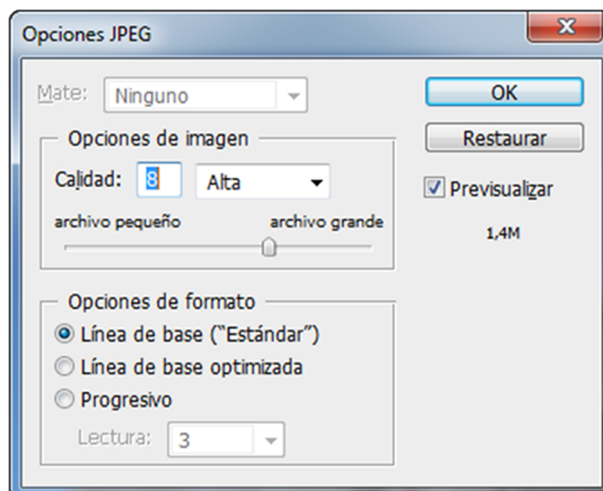


Il·lustració 21. Imatge codificada només amb el component de contínua de cada bloc

Qualitat d'imatge i taules de quantificació

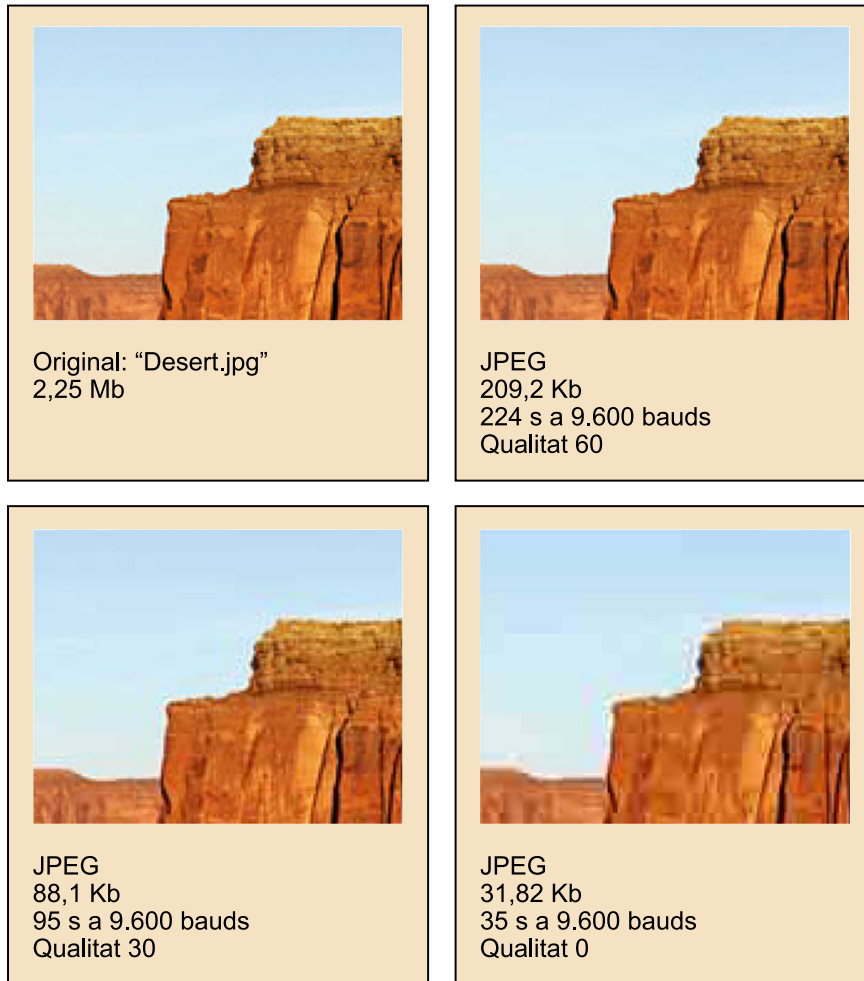
Quan es vol codificar una imatge en format JPEG, el programari (per exemple, Photoshop) pregunta el nivell de qualitat que volem mantenir de la imatge original en la nova imatge JPEG.

Il·lustració 22. Opcions de compressió JPEG de Photoshop



En general és un paràmetre quantitatiu (com un valor de 0 a 10 o de 0 a 12, o un percentatge de 0 a 100, etc.), i el primer és la pitjor qualitat i màxima compressió, i l'últim, la màxima qualitat i mínima compressió. En altres ocasions possibilita seleccionar la qualitat o compressió mitjançant valors qualitatius (de "menys detall" a "més detall"), com per exemple, en les configuracions de les càmeres digitals.

Il·lustració 23. Quadre comparatiu de compressions JPEG sobre una mateixa imatge



Aquest paràmetre de qualitat seleccionat per l'usuari està molt lligat a la taula de quantificació que s'hagi d'aplicar per a eliminar els coeficients d'alta freqüència de tots els blocs transformats de la imatge. Així, com més alt és el valor de qualitat, més es redueixen els valors de la taula de quantificació, de manera que la divisió dona un resultat més gran i "sobreviuen" més coeficients d'alta freqüència. Quan la qualitat és màxima, els valors de la taula de quantificació són tots 1, i no afecten en absolut el resultat.

El valor de qualitat també pot permetre seleccionar el codificador entre les tècniques de submostratge de color 4:2:2 o 4:2:0 per a reduir sensiblement el nombre de blocs de color i millorar la compressió sense que l'usuari hagi de fer cap acció.

Pas 5. Codificació entròpica dels coeficients

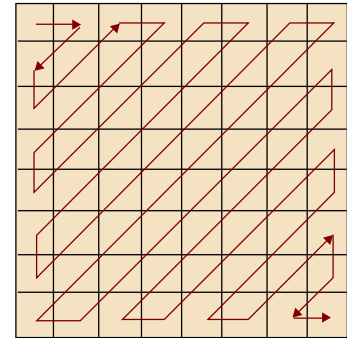
Si bé els 64 coeficients són enters, en aquest pas es converteixen a binari i s'emmagatzemen en fitxer mitjançant una codificació que redueix la necessitat de bits per a fer-ne l'emmagatzematge. Com que el coeficient de contínua és tan important, es tracta amb una codificació específica.

Primer de tot s'ordenen seguint un ordre en ziga-zaga establert (il·lustració 24).

Seguint el nostre exemple, la ziga-zaga dóna com a resultat:

15, 3, 3, -4, 0, 2, -13, -1, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, ..., 0

Tot seguit es codifiquen de la manera següent:



Il·lustració 24. El camí en ziga-zaga d'ordenació dels coeficients

a) Coeficient de baixa freqüència (de contínua): guardar la diferència entre dos valors ocupa gairebé sempre menys espai que guardar els dos valors, sempre que aquests valors siguin semblants. Això és el que s'anomena **codificació diferencial (DPCM)**, i s'aplica al coeficient de contínua de cada bloc: com que en la majoria d'imatges el valor mitjà d'un bloc és semblant al del bloc immediatament anterior, s'emmagatzema el valor del primer i després les diferències de cadascun amb el següent coeficient de contínua. Si a més s'emmagatzema en el fitxer JPEG abans que la resta de coeficients, tindrem la imatge mosaic de manera immediata com a vista preliminar. Per exemple, el coeficient de contínua d'aquest bloc és 15; si el del bloc anterior fos 23, el que emmagatzemaríem seria 7, la diferència entre l'un i l'altre.

b) Coeficients d'alta freqüència (d'alternativa): en la sèrie seqüencial de valors generada per l'ordenació en ziga-zaga acostumen a aparèixer molts valors repetits, la majoria zeros, de manera que s'hi apliquen tècniques de compressió entròpica que aprofitin aquesta característica (**codificació Huffman** o, modernament, **codificació aritmètica**).

Els coeficients d'alta freqüència es codifiquen de manera que al costat seu els precedeixi el nombre de zeros que tenen a l'esquerra. Així, la nostra sèrie de 63 coeficients queda només com a:

(0,3) (0,3) (0,-4) (1,2) (0,-13) (0,-1) (2,-1) (6,1) (6,1) EOB

en què EOB és un marcador que indica "fi de bloc".

Ara, si és una codificació Huffman, es fa servir una taula a la qual assigna el conjunt mínim de bits que calen per a emmagatzemar cada nombre:

| Categoria | Valor del coeficient | Representació en bits |
|-----------|----------------------------|--|
| 1 | -1, 1 | 0, 1 |
| 2 | -3, -2, 2, 3 | 00, 01, 10, 11 |
| 3 | -7, -6, -5, -4, 4, 5, 6, 7 | 000, 001, 010, 011, 100, 101, 110, 111 |
| 4 | -15, ..., -8, 8, ..., 15 | 0000, ... 0111, 1000, ..., 1111 |
| 5 | -31, ..., -16, 16, ..., 31 | 00000, ... 01111, 10000, ... 11111 |

| Categoria | Valor del coeficient | Representació en bits |
|-----------|----------------------------------|--|
| 6 | -63, ...-32, 32, ..., 63 | 000000, ..., 011111, 100000, ... 111111 |
| 7 | -127, ..., -64, 64, ..., 127 | 0000000, ..., 01111111, 1000000, ..., 11111111 |
| 8 | -255, ..., -128, 128, ..., 255 | 00000000 ... 11111111 |
| 9 | -511, ..., -256, 256, ..., 511 | 000000000 ... 111111111 |
| 10 | -1023, ...-512, 512, ..., 1023 | 0000000000 ... 1111111111 |
| 11 | -2047, ..., -1024, 1024, ...2047 | 00000000000 ... 11111111111 |

El nombre dret de cada parella es codifica mitjançant aquesta taula i s'hi indica també la categoria, i l'EOB es converteix sempre a (0,0):

(0,2,11)(0,2,11)(0,3,011)(1,2,10)(0,4,0010),(0,1,0)(2,1,0)(6,1,1)(6,1,1)(0,0)

Els dos primers valors sempre es poden representar amb 1 byte: els 4 bits més alts d'aquests primers valors s'encarreguen de guardar el nombre de zeros previs, i els 4 bits baixos, la categoria del valor. Es fa una taula de Huffman a partir de certa política de freqüències d'aparició, es redueix aquest valor de 8 bits de cada parella i es manté el tercer valor. Per exemple:

(0,2,11) ⇒ 00 11
 (0,2,11) ⇒ 00 11
 (0,3,011) ⇒ 010 011
 (1,2,10) ⇒ 0011 10

Mentrestant, el coeficient de contínua que ha quedat reduït a una diferència de valor 7, també es codifica amb aquesta taula; per exemple, resulta 11110.

La sèrie final que s'emmagatzema en el fitxer JPEG és 11110, 0011, 0011, 010011, 001110, etc.

Segons com s'emmagatzemin en el fitxer els components de color, es parla de **components entrelaçats** (en el qual es van emmagatzemant els blocs de luminància i de color alternativament) o **components no entrelaçats** (primer s'hi emmagatzemen els de luminància i després els de color).

Descodificació d'una imatge JPEG

El fitxer JPEG té tota la informació necessària per a recuperar la imatge comprimida segons el valor de qualitat amb què es va generar; en efecte, no inclou només els coeficients transformats de tots els blocs, sinó la informació necessària per a recuperar-los a partir de les seves versions codificades, el valor de qualitat seleccionat i la taula de quantificació seleccionada per a eliminar elements.

2.1.2. Altres modes de treball de JPEG

El mode seqüencial base que hem explicat és vàlid per a la majoria d'aplicacions de tractament d'imatge que tenen 8 bits per cadascun dels components (RGB) i fins i tot una quarta capa de transparència.

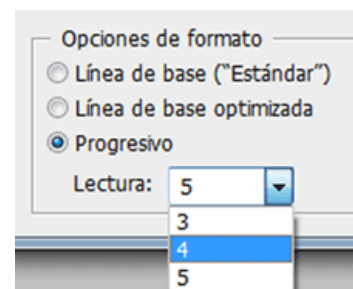
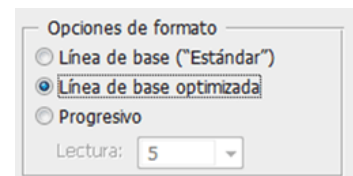
Hi ha, però, altres modes (fins a 44) per a aplicacions específiques; per exemple, per a imatges de qualitat molt alta de 12 bits per component, o per a aplicar altres sistemes de codificació de coeficients més efectius (codificació aritmètica en comptes de Huffman).

a) Mode seqüencial base optimitzat

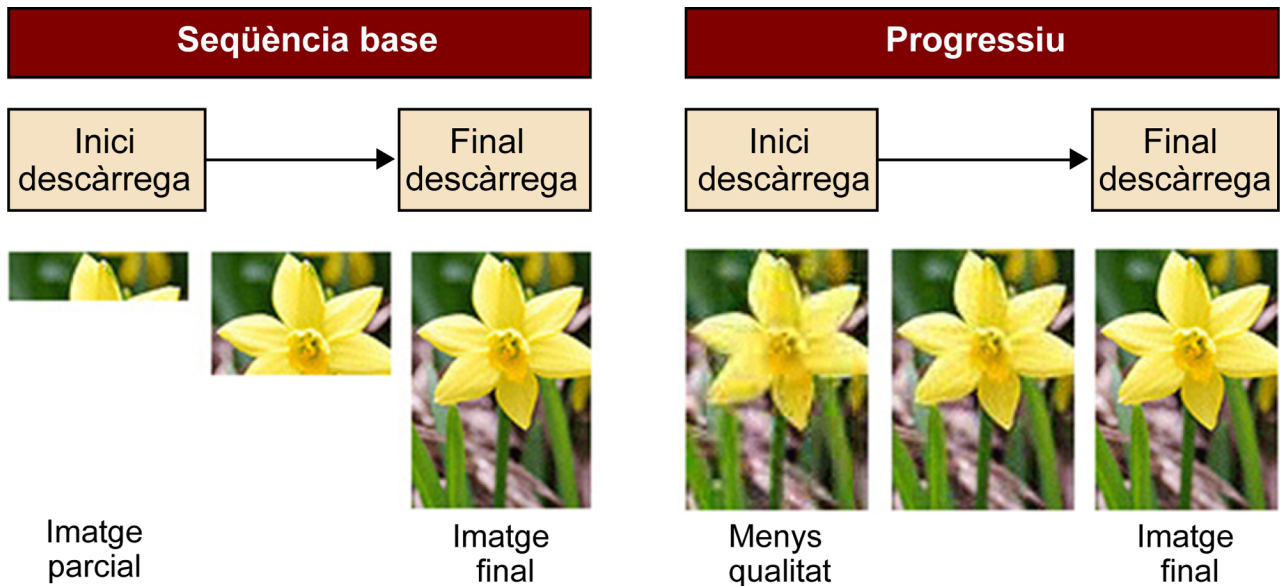
Una versió millorada del mode seqüencial base, el **mode seqüencial base optimitzat**, permet una millora del 2% al 8% en la mida del fitxer final. Hi ha algunes aplicacions, però, que pot ser que no siguin compatibles amb aquest mode.

b) Mode progressiu

El segon mode interessant amb pèrdues, el **mode progressiu**, molt utilitzat per a imatges de llocs web en què els servidors tenen connexions de baixa velocitat o tendeixen a estar molt saturats, possibilita veure una imatge mosaic (és a dir, la imatge amb només el coeficient de contínua de cada bloc) i les successives millores de qualitat en funció de l'ordre de recepció de les dades, de manera que l'espectador pot tenir vistes parcials cada vegada més detallades de la imatge que va rebent.



Il·lustració 25. Seqüència de baixada d'una imatge JPEG seqüencial base (esquerra) i la mateixa imatge codificada en mode progressiu (dreta)



El temps de baixada és aproximadament el mateix, però l'experiència d'espera de l'usuari és diferent.

Hi ha diverses maneres d'implementar aquesta solució, encara que la més generalitzada és emmagatzemar en el fitxer JPEG primer els coeficients de contínua i els primers coeficients d'alterna de tots els blocs, després coeficients d'alterna de menys rellevància visual (més detall), posteriorment de rellevància encara més petita (i més detall), etc. El procés acostuma a elaborar tres, quatre o cinc fases de separació dels coeficients. Un fitxer JPEG en mode progressiu sol tenir una mida en bytes superior que la del seu equivalent en mode seqüencial o seqüencial optimitzat.

c) Mode jeràrquic

El tercer mode amb pèrdues és el jeràrquic. El més complex, perquè proveeix de representacions progressives en resolució de la mateixa imatge, i requereix més espai. En essència, comprimeix la imatge en múltiples resolucions de manera que es pot accedir a la imatge de més baixa definició sense haver de descomprimir la imatge per complet.

A partir de la imatge original, la filtra i submostreja en els múltiples de 2 que es vulguin, codifica la imatge reduïda així en mida, descodifica aquesta imatge reduïda i interpola una segona imatge en un factor de 2 horitzontalment i verticalment, tot fent servir el mateix algorisme de descodificació que usaria el receptor. Aprofita aquesta imatge com la predicció de l'original i codifica la imatge diferència (és a dir, l'error). Després torna al principi i repeteix el procés.

d) Mode sense pèrdues

Per acabar, el **mode JPEG-LS** (JPEG sense pèrdues⁶) és un mode creat al cap d'uns anys (1993) que codifica la imatge sense pèrdues. Utilitza una DCT per a codificar, però no fa servir taules de quantificació, sinó que fa previsions

⁽⁶⁾En anglès, *lossless JPEG*.

dels píxels a partir dels píxels veïns. Destinat a aplicacions mèdiques d'imatge, aconseguix uns factors de compressió de 2:1, i no té rellevància fora d'aquest camp.

2.2. JPEG 2000

Al cap d'una dècada de l'aparició de JPEG, l'escenari havia canviat una mica: s'havien investigat amb èxit noves transformacions que superaven la clàssica DCT encara que obligaven a un càlcul intensiu, però la capacitat de càlcul dels equips domèstics i professionals quintuplicava el de feia deu anys i possibilitava un nou format de codificació que donés resposta a les noves exigències del mercat, entre les quals hi havia les següents:

- Codificació amb pèrdues o sense pèrdues.
- Factor alt de compressió amb menys artefactes visibles.
- Robustesa contra errors per a poder transmetre a baixa velocitat, per exemple.
- Capacitat d'afegir dades addicionals o metadades.
- Codificació més bona d'imatges sintètiques o bitò.

Per desgràcia, la situació legal del format JPEG 2000 sempre ha estat controvertida, ja que hi ha patents sobre parts del procés, cosa que des del principi n'ha frenat la popularització.

JPEG dona qualitats subjectives d'imatge molt baixes en compressions de menys de 0,25 bpp, en què JPEG 2000 encara pot donar qualitat fins i tot fins a 0,1 bpp. I en els rangs de compressió habituals, aquest nou sistema de codificació supera JPEG.

El secret és la **transformació ondeta**⁷, que tracta la imatge com un tot (no la descompon en blocs), hi aplica filtres espacials i aprofita de manera molt eficient la codificació dels resultats. Segons els filtres que s'utilitzin, aquesta transformació pot ser sense pèrdues o amb pèrdues, i en aquest darrer cas millora la compressió a costa d'algun efecte visual en els contorns (l'alta freqüència espacial).

Així, si JPEG 2000 és capaç de comprimir amb qualitat a una taxa més alta, és més bo per a transmetre imatge a baixa velocitat. A més, com que el tipus de descodificació és progressiu –la imatge rebuda es descodifica a partir d'una primera versió bàsica (baixa freqüència espacial) que millora detalladament–, resulta perfecte per a transmetre: si s'interromp en un moment concret, almenys tindrem una imatge completa parcialment detallada.

L'estàndard possibilita incloure dades addicionals (metadades) en el fitxer d'imatge. JPEG ja ho possibilitava en el format EXIF, però JPEG 2000 ho fa de manera més estandarditzada seguint el format XML.

Patents sobre la compressió JPEG 2000

Bona part del programari de tractament d'imatges no inclou les parts del procés patentades, o ho fa opcionalment, com Adobe Photoshop. En el cas de formats de compressió sense pèrdua, avui dia el més estès és PNG.

⁽⁷⁾En anglès, *wavelet*.

2.3. JPEG XR

El 2006, Microsoft va introduir un codificador propietari d'imatge anomenat primer HD Photo adreçat només al sector de la fotografia digital que, des del 2009, any en què es va convertir en estàndard ISO, és conegut com a JPEG XR⁸. L'extensió dels fitxers d'imatge és .wpd.

⁽⁸⁾De l'anglès, *extended range*.

Les característiques principals d'aquest codificador són les següents:

- Codificació amb pèrdues o sense pèrdues.
- Descodificació progressiva.
- Compressió més bona que JPEG.
- Un ventall de formats ampli: 8, 16 o 32 bits per canal de color –els píxels poden tenir valor enter o fins i tot flotant per a adaptar-se a càmeres fotogràfiques de rang dinàmic molt alt–, i un format en blanc i negre de 16 bits.
- El format contenidor pot incloure metadades i un canal alfa.
- No està destinat per a fer-lo servir en imatges sintètiques o amb bitons (text o gràfics).

Si bé amb aquest format només hi va ser compatible Windows a partir de la versió Vista i hi havia alguns complements⁹ comercials (per exemple, per a Adobe Photoshop oferts per Microsoft), l'avenç que té és lent enfront de l'estàndard obert d'imatge DNG (Adobe, 2004). L'alliberament de JPEG XR el 2010 per part de Microsoft li va voler donar una possibilitat d'èxit perquè en facilitava l'ús en aplicacions de codi obert, però encara és un tema obert.

⁽⁹⁾En anglès, *plugins*.

2.4. Tendències futures de la codificació d'imatge

A mitjan anys vuitanta el matemàtic Michael Barnsley va presentar la base d'un sistema de compressió d'imatges amb pèrdues basat en fractals que va generar una gran expectació.

Un fractal és un element senzill que, iterat sobre si mateix unes quantes vegades, genera un nou element complex. La idea de Barnsley era generar una imatge original a partir d'un element visual bàsic mitjançant iteracions. Així, només s'hauria d'emmagatzemar o transmetre el fractal o grup de fractals, no la imatge. Fins i tot, segons aquesta idea, es podia reconstruir la imatge original a resolucions superiors a les de partida amb molt poca capacitat de càlcul; és a dir, era un mètode independent de la resolució anomenat **escalat fractal**.

En compressions normals, la compressió fractal aconsegueix resultats semblants als dels algorismes basats en la DCT com, per exemple, JPEG, amb ràtios de 50:1. Amb compressions més grans, supera en qualitat la DCT. És curiós que la codificació fractal millora amb la riquesa visual de la imatge (amb una més alta freqüència espacial).

El problema principal consistia a trobar d'una manera senzilla el fractal o element visual bàsic (o el conjunt de fractals bàsics) que ho pogués aconseguir. La necessitat d'una gran capacitat de càlcul per aconseguir deduir-los i la falta d'un algorisme eficient per a tot tipus d'imatges ha frenat fins ara aquest sistema de compressió, encara que la recerca continua oberta fins i tot per a fer-ne l'aplicació en vídeo.

2.5. Exercicis

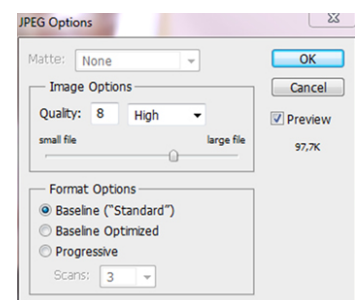
Exercici 4

Comprimiu una imatge en format JPEG mitjançant Photoshop per a veure'n l'efecte en la qualitat de la imatge i en la mida del fitxer resultant. Per a fer-ho, obriu l'aplicació i llegiu una imatge. Tot seguit, mitjançant l'opció del menú *Arxiu > Desa*, seleccioneu un nom nou i trieu com a format d'enregistrament JPEG (*.JPG, *.JPEG, *.JPE, les tres extensions habituals que s'afegeixen al nom del fitxer). Després d'acceptar, apareix una altra finestra, que us demana la qualitat visual en què voleu comprimir-ho (un valor de 0 a 12). De moment, manteniu com a opció el mode *baseline* (estàndard), del qual parlarem en l'apartat següent.

Després d'acceptar, Photoshop emmagatzemarà el fitxer comprimit. Visualitzeu el resultat i mitjançant el sistema operatiu que tingueu (Windows, OSX, etc.) preneu nota de la mida en kilobytes del fitxer resultant. Es pot calcular amb facilitat el factor de compressió.

Orientació

Aquest exercici també es pot fer amb l'aplicació de programari lliure Gimp.



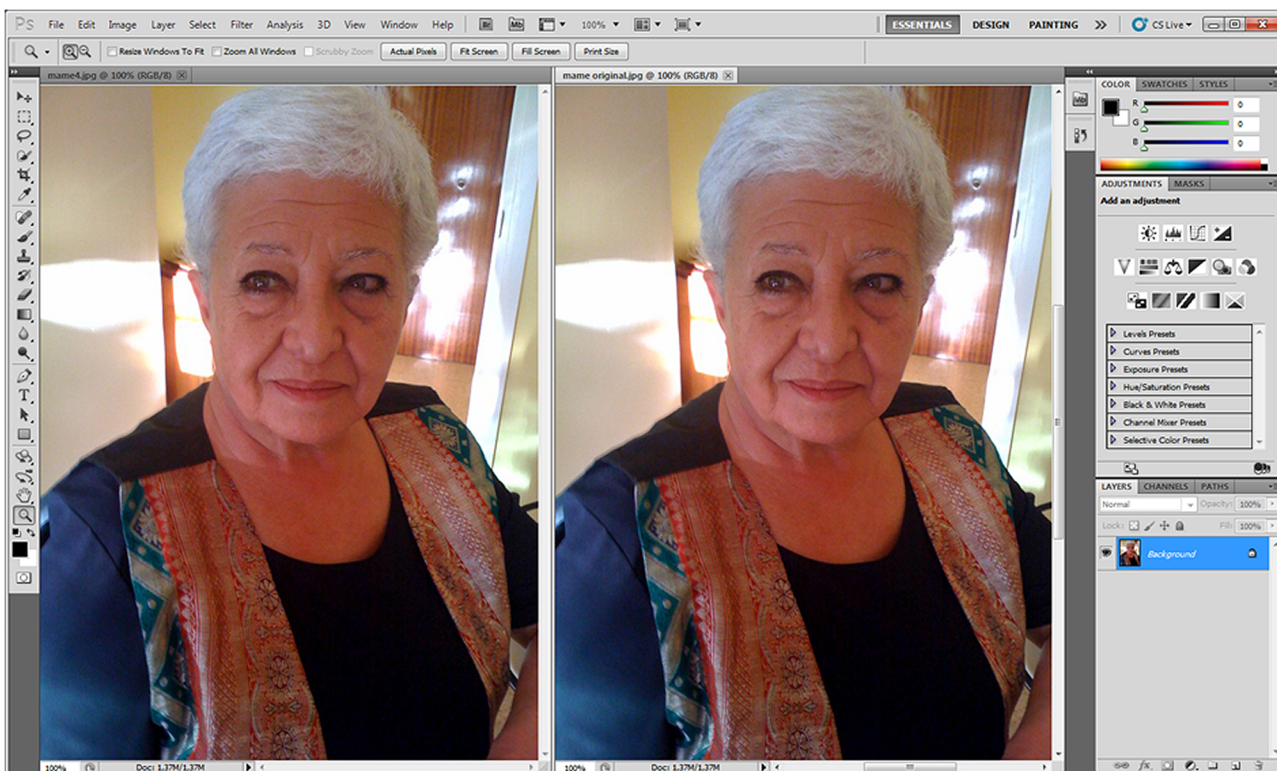
Exercici 5. Com es crea una imatge diferència amb Photoshop

El factor de compressió, com podeu veure, pot donar una idea aproximada del resultat, però no fa gaire servei si voleu saber de veritat on es genera la pèrdua d'informació visual. Per això podeu crear una imatge diferència que sigui la resta píxel a píxel entre la imatge sense comprimir i la comprimida; en aquest cas, en JPEG.

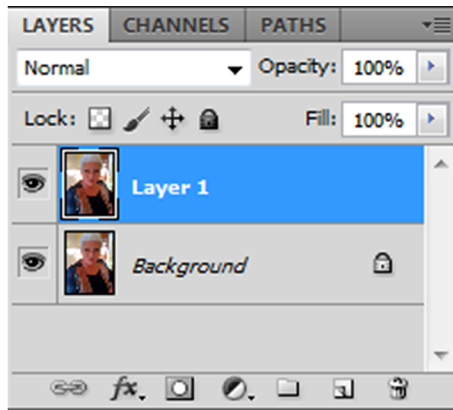
Obriu Photoshop, llegiu la imatge original i també la imatge comprimida (us aconsellem que trieu una imatge comprimida en un factor de 4 a 6, perquè així els errors seran més evidents).

Orientació

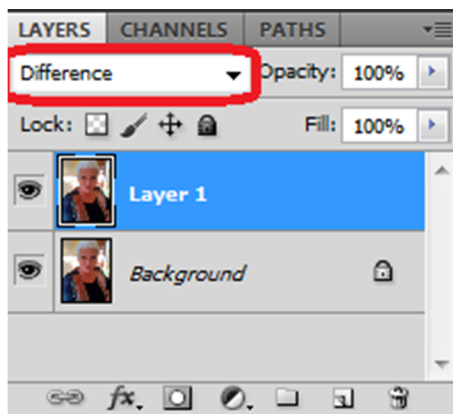
Aquest exercici també es pot fer amb l'aplicació de programari lliure Gimp.



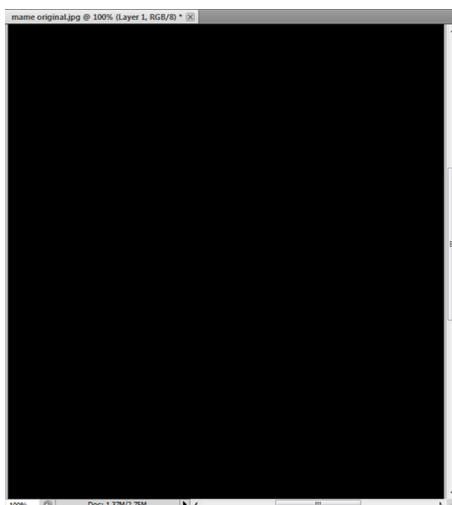
Tot seguit, seleccioneu amb un clic la imatge original sense comprimir i mitjançant *Ctrl-J* (Windows) o *Command-J* (Mac) dupliqueu-ne la capa *Fons* (o *Background*, segons l'idioma del programa) en la paleta de capes. El resultat són dues capes: la capa *Fons* original a baix i una còpia que Photoshop anomena *Capa 1* (*Layer 1*).



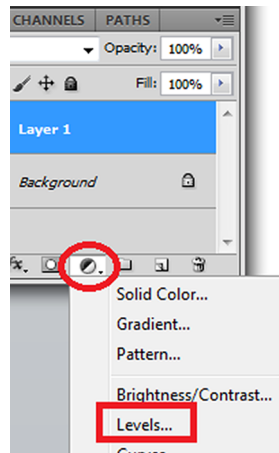
Aquests dues capes són idèntiques. Com a demostració, presentarem la imatge diferència: podem avançar que, si totes dues són iguals, la diferència entre píxels corresponents serà 0, de manera que tots els píxels de la imatge diferència seran de valor 0, és a dir, totalment negres. En el desplegable de la finestra de capes, trieu *Diferència (Difference)*:



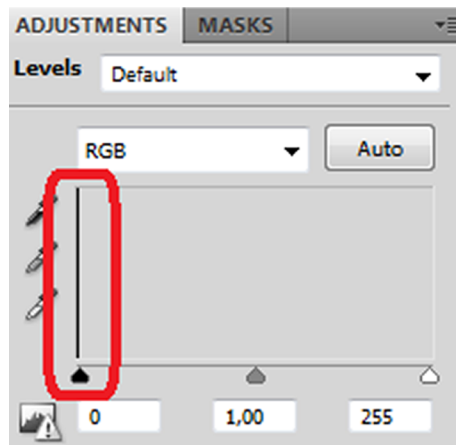
Tal com era previsible, la **imatge diferència** apareix totalment negra.



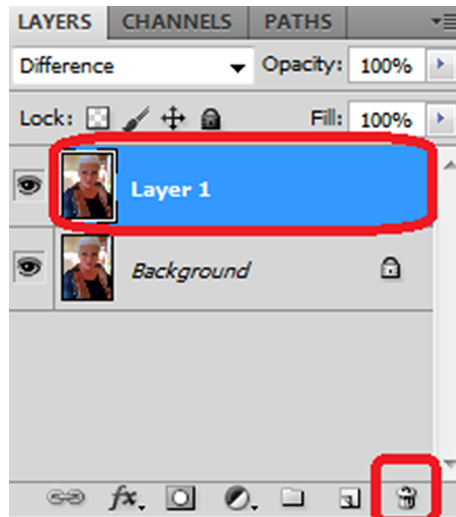
Siguem escèptics, i confirmem que és així d'una manera més explícita. Per a fer-ho, seleccioneu l'eina d'ajust i, entre les opcions de què disposeu, trieu *Nivells (Levels)*.



Activarà una finestra de diàleg en la qual es reflecteix l'histograma de la imatge diferència. És tan estret perquè tots els píxels són del mateix valor, no es distribueixen en altres valors que no siguin zero.



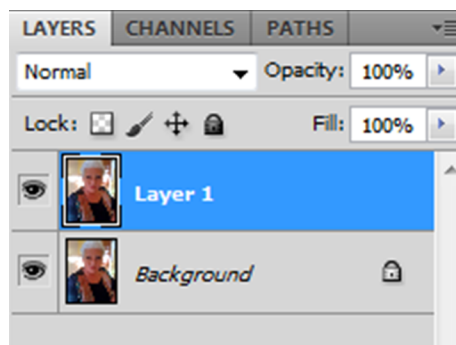
Un cop hem demostrat que la imatge diferència entre dues imatges iguals és negra, esborreu la *Capa 1 (Layer 1)* seleccionant-la amb un clic a la finestra de capes i després fent clic a la paperera situada en aquesta mateixa finestra.



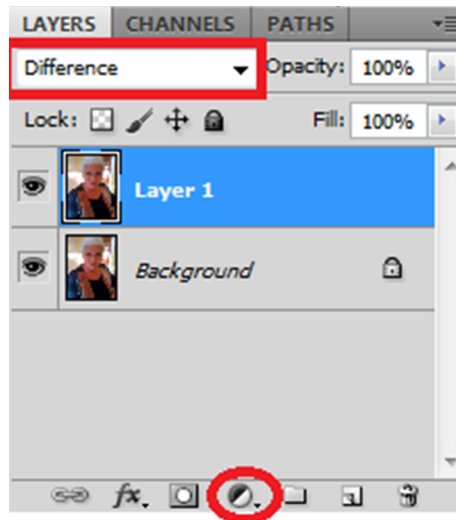
Ara que ja sabem confeccionar una imatge diferència i que hem tornat al punt de partida, torneu a fer l'operació però amb les dues imatges que encara eren obertes a Photoshop. Seleccioneu l'eina *Moure* a la barra d'eines, cliqueu amb el ratolí sobre la **imatge comprimida** sense deixar-lo anar, arrossegueu el ratolí fins a la imatge sense comprimir, pitgeu i mantingueu pitjada la tecla *Majús*, i finalment deixeu anar el ratolí. Si bé no sembla que visualment hi hagi hagut cap efecte, a la finestra de capes veureu que s'ha creat una segona capa, *Capa 1* o *Layer 1*, que és la imatge comprimida, mentre que *Fons*, o *Background*, és la imatge sense comprimir.



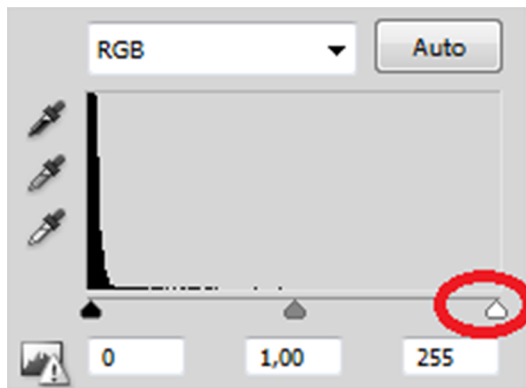
Eina Moure



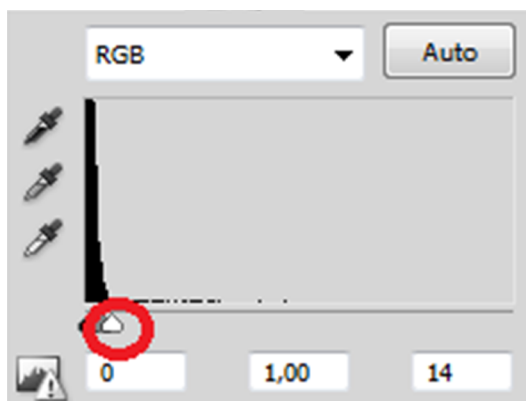
Ja tenim les dues imatges en capes, de manera que podeu tornar a fer l'operació *Diferència* (apareixerà, aquesta, més o menys negra) i, tot seguit, l'eina de capes, triant *Nivells (Levels)*:



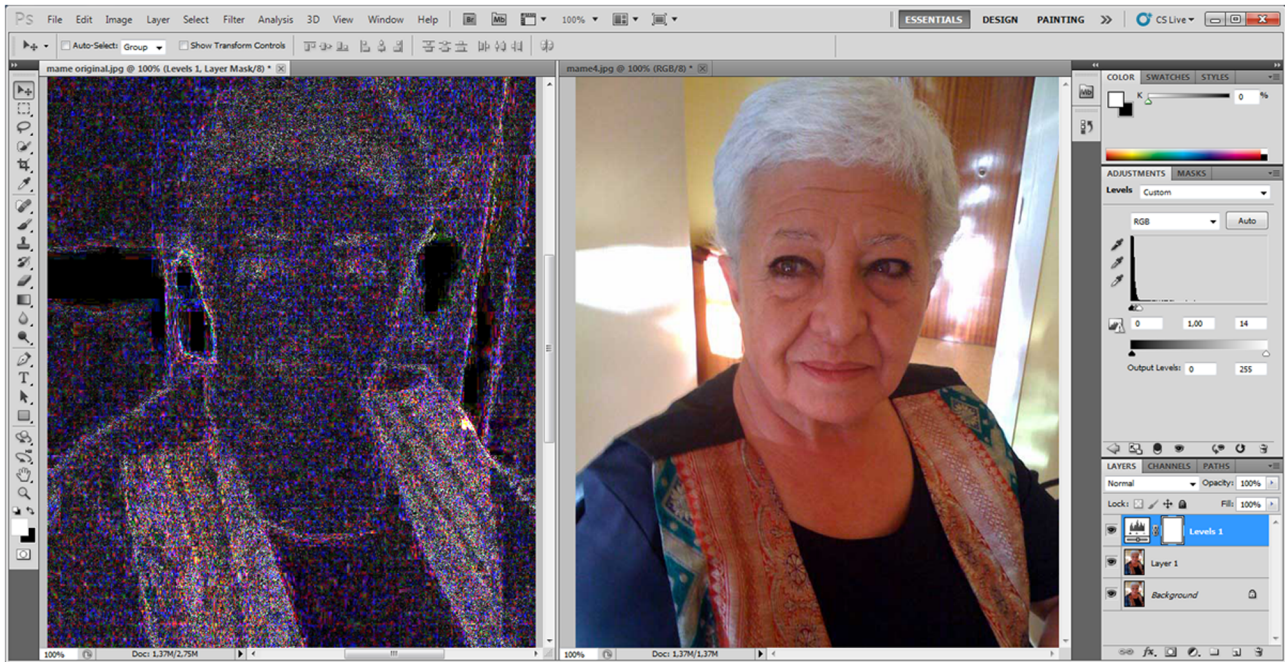
Gairebé amb seguretat, l'histograma no serà tan uniforme:



Per a poder destriar amb claredat els píxels diferència, desplaceu l'*slider* blanc cap a l'esquerra i així realçareu visualment els valors.



Així, tots els errors generats per la codificació amb pèrdues de JPEG es reflecteixen ben situats en zones específiques de la imatge, com les vores dels objectes i en general les zones amb gran detall visual.



Fent el procés amb nivells de compressió més elevats apareixen igualment errors en zones de color homogeni i un clar efecte de blocs en general, intrínsecs a la manera com JPEG treballa la imatge.

Exercici 6

Visualitzeu la imatge diferència en el cas de compressió JPEG en mode de línia de base optimitzada (*Baseline optimized*) i valoreu la diferència amb JPEG estàndard (*Baseline*) utilitzant la mateixa imatge original de l'exercici anterior i comprimint-la a la mateixa qualitat. Valoreu la qualitat visual i el factor de compressió de la resultant.

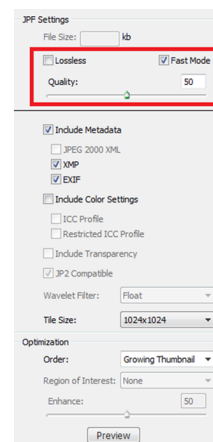
Exercici 7

Feu l'exercici anterior triant en aquest cas el mode progressiu (*Progressive*) amb quatre lectures. Mitjançant un navegador o programa de visualització, obriu el fitxer d'imatge que s'ha generat i comproveu que es visualitza amb una millora gradual.

Exercici 8

Photoshop llegeix i enregistra en format JPEG 2000 (extensions *.JPF, *.JPX, *.JP2, *.J2C i *.JPC) i accepta una configuració personalitzada en cada enregistrament. Entre les opcions de personalització de què disposa, destaquem si voleu una compressió sense pèrdues (*lossless*) en una qualitat indicada per un valor de 0 a 100.

Feu unes quantes compressions amb aquest format amb el mode amb pèrdues, proveu el format sense pèrdues i valoreu el nivell de compressió assolit pel que fa a la mida del fitxer.



Il·lustració 26. Opcions de codificació amb pèrdues o sense de JPEG 2000

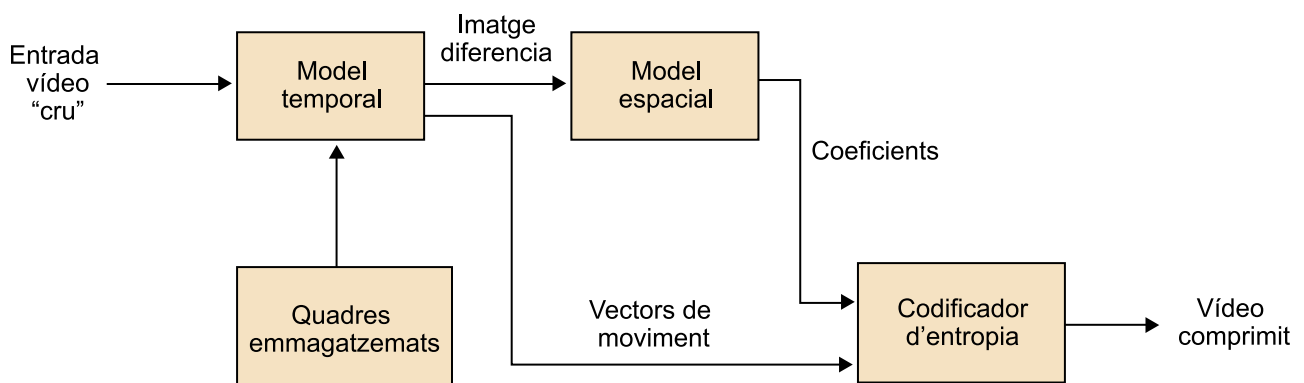
3. Codificació de vídeo i àudio

Vista la codificació d'imatge fixa, els mateixos conceptes de *redundància espacial*, *temporal* i *psicovisual* es poden aplicar a àudio i vídeo, fins i tot amb més èxit. Les metodologies per a aprofitar aquestes redundàncies, però, són més complexes si es té en compte la complexitat d'aquest tipus d'informació i el volum de dades per segon que genera.

3.1. Models de codificació d'àudio i vídeo

La codificació amb pèrdues impera en gairebé tots els estàndards de compressió que hi ha per a aquests models, perquè són els únics que avui dia aconseguen ràtios de compressió interessants. Bona part d'aquests models es poden representar amb l'esquema següent, que representa un **codificador genèric de vídeo amb pèrdues (codec)**:

Il·lustració 27. Model bàsic d'un codificador de vídeo amb pèrdues



La família MPEG, H.264, Theora, VC-1, etc., són exemples de còdecs que fan servir aquest model, i realitzen amb més o menys gràcia cadascuna d'aquestes etapes:

a) La primera fase és el **model temporal**, el qual, a partir dels quadres emmagatzemats del vídeo llegits fins llavors, "preveu" el quadre següent que arribarà per l'entrada de vídeo "cru". Aquesta previsió no acostuma a estar especificada i depèn de cada fabricant, cosa que condiciona bona part de la qualitat final de tot el procés. Per exemple, un codificador MPEG-2 de l'empresa Tandberg té un plantejament diferent de previsió del quadre següent que el que té el codificador MPEG-2 de l'empresa Thomson. Per descomptat, el funcionament del model temporal de cadascuna d'aquestes empreses és un secret ben guardat.

b) La fase següent se centra en el **model espacial**. La imatge diferència es comprimeix utilitzant la similitud dels seus píxels: "comprimir una diferència" seria bàsicament un resum en tres paraules de tot el procés. Cada sistema de

compressió de vídeo fa servir aquí les seves pròpies tècniques de compressió d'imatge, encara que les que utilitza, per exemple, H.264 no disten gaire de les que estima JPEG. Sens dubte, la majoria amb pèrdues. I mantenint aquesta idea al cap, igual que JPEG, el resultat és una sèrie de valors binaris que també s'anomenen *coeficients espacials* i que fan el mateix paper: representen la imatge resultant amb un altre vocabulari matemàtic.

c) Finalment, collant una mica més la rosca, tant els coeficients de la imatge com els vectors del moviment que s'hi han detectat són empaquetats de manera eficient en el **codificador d'entropia**, que analitza la freqüència d'aparició d'aquests coeficients i vectors, i hi assigna menys bits com més freqüents són, de manera que redueix la mida final del fitxer o del flux de dades de transmissió¹⁰ al mínim possible (o al valor més pròxim a aquest mínim).

⁽¹⁰⁾En anglès, *stream*.

El resultat és una sèrie binària compacta, que conté tota la informació necessària per a poder-la descomprimir. Els sistemes de descompressió estan ben definits, de manera que en aquest cas no hi ha cap diferència entre descodificadors d'una empresa o una altra.

Exemples d'estàndards

Avui dia els dos estàndards de còdecs més habituals en els sistemes d'IPTV són H.264 (anomenat també *MPEG-4 Part 10* o *AVC*), desenvolupat en conjunt entre UIT-T i MPEG; i VC-1 (nucli de Windows Media Audio i Windows Media Video 10), desenvolupat per Microsoft. Des del 2010, Google TV, amb el còdec VP8, intenta fer-se un lloc en el mercat donant suport sense fissures a WebM.

Per als sistemes que fan servir xarxes basades en IP, com Internet i les xarxes locals, o serveis IP sobre 3G com a mitjà de transport, les compressions més utilitzades són les següents:

a) **MPEG-2, H.264/MPEG-4 AVC i 3GPP**: són formats estàndards utilitzats sobretot per QuickTime en la línia de servidors Darwin / QuickTime Stream Server i en el reproductor QuickTime. Adobe, també, en el Flash Video Server, fa servir H.264 com a còdec a partir de la versió 9. Cal destacar que l'ús dels estàndards MPEG requereix el pagament de llicència tant en servidors com en reproductor. Per la seva qualitat, H.264/MPEG-4 també es fa servir en emmagatzematge de vídeo en Blu-ray i en les emissions HD per antena, cable o satèl·lit, mentre que MPEG-2 ha quedat relegat com a estàndard d'emmagatzematge per a DVD i emissions digitals en qualitat estàndard.

b) **Windows Media**: és una codificació propietària desenvolupada per Microsoft; té la seva fortalesa en la transmissió en temps real de baixa amplada de banda (com les d'Internet). Pot ser servit per la línia de servidors Microsoft utilitzant el servei Windows Media Services i reproduït per Windows Media Player. Aquesta codificació requereix un permís especial i un pagament de llicència a Microsoft per a poder-la fer servir en una altra línia de servidor/reproductors, de manera que l'ús que se'n fa està en declivi enfront d'H.264 o VC-1.

c) **VP8**: és el còdec obert de vídeo per a transmissió en temps real de WebM, la proposta de format de Google per a utilitzar-lo en HTML 5. Té una qualitat semblant a H.264 i es basa en una biblioteca de programari lliure anomenada *x264* amb llicència GNU. El contenidor de VP8 es basa en el contenidor obert Matroska.

d) **Theora**: és un format obert, el principal atractiu del qual és que no requereix el pagament de llicència per a fer-lo servir. Presenta una gamma d'estàndards en desenvolupament continu que vol competir amb la línia d'estàndards MPEG-4. La codificació de vídeo Theora està basada en la codificació On2's VP3, i el seu contenidor més conegut per a emmagatzematge és el format Ogg.

e) **RealVideo**: és una codificació propietària, desenvolupada per RealNetworks, empresa pionera en transmissió de dades d'àudio i vídeo en temps real a Internet. Actualment ha caigut en desús. La fortalesa d'aquesta codificació estava en la transmissió en temps real



de baixa amplada de banda. Podia ser servida per la línia de servidors Helix i reproduïda pel RealPlayer (o els precedents HelixPlayer i PlayerOne), tots també de l'empresa RealNetworks. Aquesta codificació requeria un permís especial i un pagament de llicència a RealNetworks per a poder-la fer servir en una altra línia de servidor/reproductors, de manera que el cost d'aquest còdec i la falta d'inversió en innovació ha fet que gairebé hagi desaparegut del mercat actual.

3.2. Compressió d'àudio digital

Les tècniques de compressió d'àudio són cronològicament anteriors a les de vídeo. Com en el vídeo, hi ha tècniques amb pèrdues i sense pèrdues, algunes d'específiques per a transmetre en temps real i d'altres per a emmagatzematge.

Totes aquestes tècniques es basen en la reducció de la redundància que hi ha en el senyal per a minimitzar-ne el flux de dades generat per segon sense que en resulti afectada la qualitat de l'àudio segons les seves necessitats.

De l'ampli ventall d'estàndards que hi ha, n'hi ha que se centren en la codificació de la veu humana per a transmetre-la en xarxes digitals. Alguns serveis de transmissió de veu en temps real com ara la telefonia fixa, la telefonia mòbil o Internet assoleixen grans taxes de compressió perquè se centren en les característiques específiques d'aquest senyal. Se solen basar en tècniques de predicció (en poques paraules, es transmeten les diferències entre les mostres d'àudio previstes i les reals).

Estàndards de compressió d'àudio digital

Exemples il·lustratius:

- Les xarxes GSM i UMTS codifiquen l'àudio en format AMR, tant en l'emissió com en l'emmagatzematge de missatges a la bústia de veu (extensió .amr), i és el format base del grup 3GPP per a futures xarxes mòbils de quarta generació.
- Skype es basa en el seu propi estàndard, Silk (Super Wideband Audio Codec).

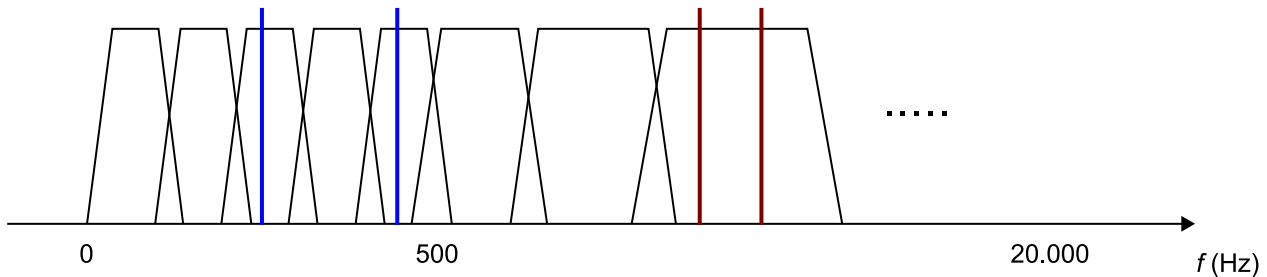
Un altre gran grup, més interessant en el nostre cas, té com a objectiu aconseguir la màxima qualitat de so amb el mínim nombre de bits per segon, fins i tot possibilitant so multicanal. La majoria d'aquestes tècniques es basen en la codificació perceptiva, un avenç en el món de la codificació d'àudio que va implementar per primera vegada Philips en el seu ja desaparegut sistema d'àudio compacte digital (DCC) que volia substituir el casset de cinta, i base de l'MP3.

La **codificació perceptiva** neix dels estudis sobre la manera com l'ésser humà interpreta el so.

Apareixen tres patrons de codificació perceptiva que obren la porta a comprimir l'àudio sense una pèrdua de qualitat audible:

1) L'orella no detecta tots els sons que rep, i es comporta com un “banc de filtres” que més o menys s'encavalquen en la freqüència. El cervell diferencia clarament dos sons situats en bancs diferents, però no ho fa si aquests dos sons són al mateix banc de freqüències.

Il·lustració 28



L'orella diferencia dos tons (en blau) situats en freqüències de bandes diferents, mentre que interpreta com un sol to dos tons situats en la mateixa banda (en vermell)

2) La sensibilitat de l'orella davant un so canvia si hi ha més sons. D'alguna manera, hi ha sons que n'emascaren d'altres i els fan indetectables per a l'orella. Aquest fenomen s'anomena **emascament freqüencial**.

3) Encara més, si mentre escoltem un so feble aquest so s'atura, tardarem una estona fins a poder tornar a sentir un so feble. Aquest segon fenomen s'anomena **emascament temporal**.

El sistema es fonamenta en les dades experimentals recollides per milers de proves en voluntaris, de manera que el model psicoacústic es pot interpretar com un comportament mitjà de l'orella humana. Per tant, hi ha subjectes acústicament més ben capacitats que d'altres, que poden detectar la pèrdua de qualitat inherent a aquesta codificació.

Dels còdecs d'àudio habituals avui dia, la majoria segueixen aquest model o variacions d'aquest model, i d'altres han tendit a sistemes de compressió més conservadors.

Còdecs d'àudio habituals

- a) El format de so **MP2** (conegut com a *Musicam*) és l'habitual en fitxers d'àudio professional.
- b) L'estàndard MPEG-1 Audio Layer 3, nascut a partir d'MP2, és un clàssic de la família de còdecs amb pèrdues. Es va definir per a emmagatzematge d'àudio digital domèstic. Es coneix com a **MP3**.
- c) **Vorbis** també és un estàndard d'àudio amb pèrdues resultat d'un projecte de programari lliure ofert com a alternativa a l'estàndard propietari MP3. Ofereix una qualitat d'àudio semblant, i fins i tot més bona a baixes velocitats de transmissió o lectura (< 64 kbps).
- d) **AAC** (Advances Audio Coding), posterior a MP3 i de més qualitat; és el que tenen els equips domèstics portàtils d'àudio d'Apple (iPod, iPhone, iPad), la família de mòbils Android, Sony (PlayStation 3), Nintendo (DSi, Wii), etc. Funciona amb àudio multicanal (fins a 48 canals) i és una codificació amb pèrdues.

e) **Dolby AC-3** (conegut com *Dolby Digital*) és un estàndard propietari amb pèrdues nascut per al cinema, però la seva alta eficiència (320 kbps per a cinc canals d'àudio d'alta fidelitat més un sisè canal de so envoltant o *surround*) i una bona relació amb els fabricants d'equips el van alçar com a estàndard d'àudio per a DVD i Blu-ray, i fins i tot per a la televisió digital als Estats Units (i, per empatia, també algunes televisions europees).

f) **Dolby Digital Life** és un estàndard de codificació en temps real adreçat al mercat multimèdia, sobretot al sector de videojocs.

g) **Windows Media Audio** (WMA) és l'aposta de Microsoft per un estàndard de codificació. N'hi ha tant amb pèrdues com sense pèrdues (WMA Lossless).

h) **Real Audio**, només com a comentari, va ser un estàndard de compressió amb pèrdues dissenyat específicament per a transmissió en temps real, molt utilitzat en els primers anys d'Internet, però la naturalesa propietària d'aquest estàndard i la falta de millores per part de l'empresa RealNetworks van fer que desaparegués ara fa uns anys.

3.3. Diferenciar contenidor i còdec

En emmagatzemar la informació audiovisual en un fitxer per a emmagatzemar-la o transmetre-la, s'utilitza un format de contenidor.

Un **format de contenidor** és un format d'arxiu digital que emmagatzema un determinat tipus d'informació codificada amb un o diversos còdecs estàndard.



Il·lustració 29

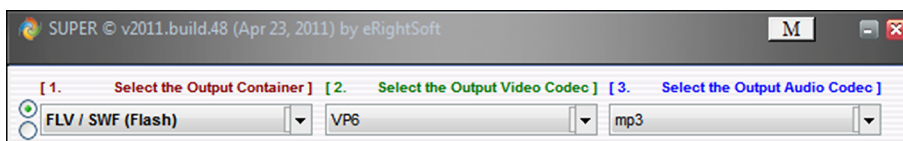
Un exemple d'associació pura entre còdec de vídeo i contenidor

És freqüent confondre tots dos conceptes i barrejar el nom del *còdec* amb el del *contenidor*. Així, Flash Video és un format contenidor que té una extensió de fitxer .flv o .f4v i en què el vídeo que pot estar codificat en H.264 o en un altre còdec, com Sorenson Spark o VP6, mentre que l'àudio pot estar codificat en MPEG-1 Audio Layer 3.

En alguns casos, el nom del contenidor acaba substituint el del còdec per costum o simplificació. Per exemple, el contenidor per al còdec MPEG-1 Audio Layer 3 és conegut comunament per *MP3*, i l'extensió dels arxius és .mp3, però se sol parlar de *codificació MP3*.

Els contenidors més complexos admeten diferents còdecs d'àudio i vídeo i són capaços de manejar subtítols, capítols i metadades (o *tags*). Un dels rols més importants del contenidor és propiciar informació temporal per a poder sincronitzar la reproducció de més d'un flux de dades simultàniament.

Il·lustració 30



Bona part de les aplicacions de codificació d'àudio i vídeo indiquen clarament els còdecs de vídeo i àudio possibles d'un contenidor seleccionat (a la imatge, Super ©)

Algunes característiques que diferencien els diversos contenidors són aquestes:

- La popularitat (i el suport per part de les aplicacions).
- La sobrecàrrega¹¹ (diferents contenidors presenten diferents mides d'arxiu per al mateix contingut).
- El suport de còdecs (per exemple, alguns contenidors com AVI no admeten codificacions amb quadres B).
- El suport de subtítols o altres característiques avançades.
- El fet que sigui apte per als servidors de reproducció en temps real com a format d'entrada.

⁽¹¹⁾En anglès, *overhead*.

La majoria dels contenidors tenen la possibilitat de fluxos de bits variables tant per a àudio com per a vídeo. Una de les excepcions més notòries és, per exemple, el contenidor AVI de Microsoft, que no ho permet. Avui dia, quan un contenidor no té alguna característica que volem que tingui, els programes configuren extensions per a poder-hi funcionar, encara que moltes vegades aquestes extensions són incompatibles les unes amb les altres, de manera que l'escenari es complica. La taula següent resumeix les característiques dels contenidors més populars:

| Contenidor | Característiques | | | |
|---|---|------------------------------------|------------------------------------|----|
| | Còdec de vídeo | Còdec d'àudio | Transmissió de dades en temps real | 3D |
| .3gp | MPEG-4 Part II H.264/MPEG-4 AVC | AAC AAC v2 | – | – |
| .avi | La majoria excepte H.264/AVC | La majoria | Sí | – |
| .divx | MPEG-4 Part II | MPEG-1 Audio AC-3 PCM | Sí | – |
| (Flash Video) .f4v | H.264/MPEG-4 AVC | AAC MPEG-1 Audio | Sí | – |
| (Flash Video) .flv | H.264/MPEG-4 AVC Sorenson Spark VP6 | AAC MPEG-1 Audio PCM | Sí | – |
| (Matroska) .mkv .mka .mks .mk3d | La majoria | La majoria | Sí | Sí |
| (MPEG) .mp4 | MPEG-2 Part II H.264/MPEG-4 AVC H.263 VC-1 | AC-3 MPEG-2 MPEG-4 Vorbis | Sí | Sí |
| (MPEG) .mpg .mpeg | MPEG-1 MPEG-2 | MPEG-1 Layer I,II,III | Sí | – |

| Contenedor | Característiques | | | |
|-------------------------------------|---|---|------------------------------------|----|
| | Còdec de vídeo | Còdec d'àudio | Transmissió de dades en temps real | 3D |
| (MPEG Program Stream) .ps | MPEG-1 MPEG-2 MPEG-4 Part II H.264 VC-1 | MPEG-1 Layer I,II,III | Sí | - |
| (MPEG Transport Stream) .ts | MPEG-1 MPEG-2 MPEG-4 Part II H.264 VC-1 | MPEG-1 Layer I,II,III | Sí | - |
| (Blu-ray Transport Stream) .m2ts | MPEG-2 Part II H.264 VC-1 | Dolby AC-3 Dolby Digital Plus DTS DTS HD | No aplicable | Sí |
| (AVID) .mxf | La majoria | La majoria | No aplicable | - |
| (XIPH.ORG) .ogg | Theora i molts més | Vorbis i molts més | Sí | - |
| (Apple) .qt .mov | MPEG-1 MPEG-2 Part II MPEG-4 Part II H.264/MPEG-4 AVC Cinepak | AAC MPEG-1 Audio PCM | Sí | - |
| (Real) .rmvb | Real Video | Real Audio AAC Vorbis | Sí | - |
| (Microsoft) .wma .wmv | La majoria excepte H.264/AVC | La majoria | Sí | - |
| (Google) WebM | VP8 | Vorbis | Sí | Sí |

4. L'estàndard MPEG-2

El grup MPEG¹² de l'ISO¹³ és el grup més destacat i acceptat per la indústria pel que fa a estandardització de vídeo digital.

⁽¹²⁾MPEG és la sigla de *Motion Picture Expert Group*.

Després de l'èxit obtingut amb l'estàndard MPEG-1 el 1993, el grup va decidir configurar un estàndard més complet i millorat adreçat a un mercat potencial atractiu i de gran escala com el de *broadcast*¹⁴ de televisió digital. Acatat el 1995, l'estàndard presenta, en relació amb el precedent, millores en la compressió, codificació entrelaçada (a més de progressiva) i una gran flexibilitat gràcies a l'ús de perfils i nivells. MPEG-2 estandarditza tant el vídeo com l'àudio que l'acompanya.

⁽¹³⁾ISO és la sigla d'*International Organization for Standardization*.



⁽¹⁴⁾O bé, difusió àmplia.

Els perfils i els nivells són variacions de qualitat de compressió que permeten adaptar l'estàndard a les exigències dels sistemes particulars, i a més ofereixen als fabricants regles més clares respecte a la conformitat dels seus productes amb l'estàndard. Els perfils i nivells d'MPEG-2 els veurem més endavant.

Avui dia l'estàndard MPEG-2 es fa servir molt, encara que és reemplaçat a poc a poc per MPEG-4 Part 10/H.264/AVC, que veurem més endavant, i que comparteix molts detalls amb MPEG-2. És el que es fa servir per als sistemes de *broadcast* de qualitat estàndard de televisió digital (DVB), tant per cable (DVB-C) com per satèl·lit (DVB-S) o terrestre (DVB-T), i és present en el mercat domèstic mitjançant el format d'emmagatzematge DVD.

Vegeu també

Els estàndards MPEG-4 Part 10/H.264/AVC s'estudien en l'apartat 5 d'aquest mòdul didàctic.

Sense ànim d'entrar en detalls massa tècnics, donarem una visió general de l'estàndard a partir de les parts que el conformen: primer, veurem l'apartat més important, la definició del format multimèdia; després, també sense entrar en detalls, veurem els fluxos de dades i la forma en què s'empaqueten.

L'emmagatzematge i la transmissió de fluxos MPEG-2 s'estudien amb més detall en altres mòduls. Aquí només farem una primera aproximació a aquest estudi. Tancarem aquest apartat amb una breu ressenya del còdec estandarditzat en MPEG-2 i una descripció dels perfils i nivells de qualitat permesos.

4.1. Antecedents

Alguns estàndards anteriors a MPEG-2 van ser una font d'inspiració per a MPEG-2, i en algun cas encara són vigents: JPEG / JPEG 2000, MPEG-1 per a vídeo CD i H.261 i H.263 per a videoconferència.

a) JPEG / JPEG 2000

Semblant al grup de treball MPEG, el JPEG¹⁵ és un grup de treball de l'ISO, en aquest cas especialitzat en la compressió d'imatges. El 1992 va acabar l'estàndard JPEG¹⁶. La compressió espacial de la codificació de vídeo digital està molt lligada a les tècniques presentades pel grup JPEG.

⁽¹⁵⁾JPEG és la sigla de *Joint Photographic Experts Group*.

⁽¹⁶⁾ISO/IEC 10918-1 / ITU-T Recommendation T.81.

El 2000 es va presentar una versió millorada de l'estàndard conegut com a JPEG 2000¹⁷ amb un nou sistema de codificació més eficient que a més possibilitava codificar de manera escalada i fins i tot amb compressió sense pèrdues. En aquell moment no va tenir tanta popularitat perquè tenia una llicència, però des del 2011 sembla que n'ha ressorgit l'ús.

⁽¹⁷⁾ISO/IEC 15444.

b) MPEG-1 per a vídeo CD

El primer estàndard del grup MPEG és l'MPEG-1¹⁸. Culminat el 1993, avui dia es continua fent servir. El cas més notori d'èxit és el còdec d'àudio MP3¹⁹.

⁽¹⁸⁾ISO/IEC 11172.

Per a vídeo, el còdec MPEG-1 es va utilitzar en el format vídeo CD (o VCD), actualment reproduïble en la majoria dels reproductors DVD i d'una qualitat semblant a la d'un vídeo VHS domèstic.

⁽¹⁹⁾Especificat en la norma MPEG-1 Part 3 Audio Layer 3.

c) H.261 i H.263 per a videoconferència

El 1993 la UIT-T estandarditza l'H.261, un còdec per a serveis de videoconferència de baixa velocitat (es transmetia per RDSI a 64 kbps o múltiples d'aquest nombre). Pensat per a xarxes commutades com la de telefonia analògica però amb serveis digitals com RDSI, avui dia és l'únic context en què es continua fent servir en general per raons de compatibilitat cap enrere.

El 1998 la UIT-T estandarditza l'H.263, un còdec de vídeo de més qualitat i menys flux de bits resultant (30 kbps). En aquest còdec apareix el concepte de *diferents perfils de codificació* (variacions de qualitat en la compressió que al principi van generar problemes de compatibilitat entre els fabricants però que es van resoldre quan es van aclarir les especificacions en l'H.264).

4.2. Què és MPEG-2

Entrem a MPEG-2. L'estàndard MPEG-2 està completament definit en l'ISO/IEC 13818. Aquesta norma està dividida en parts segons el tema. Cada part es considera un estàndard per si mateixa; per tant, de manera més correcta, MPEG-2 és un conjunt d'estàndards.

Les parts que conformen l'estàndard MPEG-2 són, entre d'altres, les següents:

- El sistema: com se sincronitzen i conjunten els fluxos de vídeo i àudio en un sol flux de dades.
- El còdec de vídeo per a senyals entrelaçats i progressius.
- El còdec d'àudio, que és una extensió de l'MPEG-1 Audio (conegut també com a MP3), capacitat per a so multicanal.
- El control del flux per part de l'usuari final (per a reproduir, aturar, situar-se en un punt concret, estructurar en escenes, etc.).

La propietat intel·lectual de l'estàndard MPEG-2 és compartida per moltes corporacions. Hi ha més de cinc-centes patents que formen l'estàndard. Els venedors de productes i serveis basats en l'estàndard MPEG-2 han de pagar per l'explotació de la llicència. La institució beneficiada és MPEG-LA, la qual administra el conjunt de patents MPEG.

4.3. Conceptes bàsics: GOP i MPEG

Quan seiem còmodament davant l'equip de música per escoltar un DVD en el reproductor o un fitxer en format MPEG-2 en l'equip multimèdia o ordinador i volem anar a un punt concret d'aquest DVD o fitxer, l'avanç ràpid o el retrocés del comandament a distància no és fluid. Abans (molt abans), quan avançàvem o retrocedíem en una cinta de vídeo com VHS o Betamax, podíem veure amb més o menys claredat la pel·lícula accelerada, però ara la imatge va fent salts de segons en segons, i presenta instantànies de la pel·lícula.

On són la resta de fotogrames? Per què no veiem els actors moure's de manera accelerada? La culpa és del GOP²⁰ (grup d'imatges).

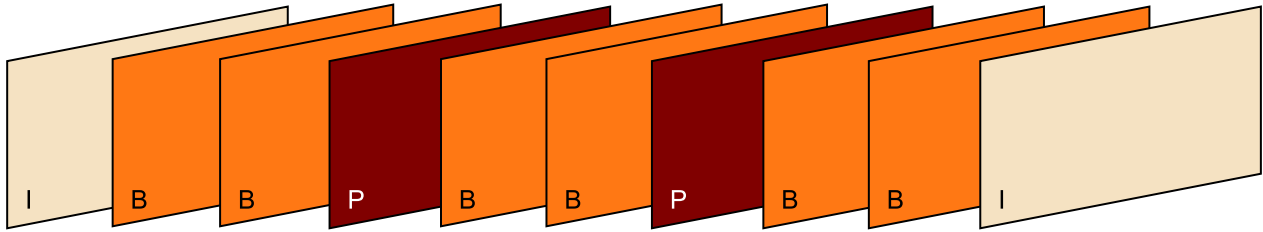
4.3.1. GOP

La imatge en moviment es pot comprimir amb més pèrdues que la fixa, perquè entre una imatge i la següent no hi sol haver gaires diferències (excepte un canvi de pla o una transició ràpida). La intenció és, de totes aquestes imatges, mantenir-ne unes quantes d'intactes (és a dir, amb una compressió semblant al JPEG, anomenada *intraquadre* o *intraframe*), que serviran de referència a la resta, que es calcularà com a diferència o desplaçament d'aquestes. Les imatges de referència són denotades amb la lletra I (d'*intracode*).

⁽²⁰⁾De l'anglès *group of pictures*.

Vegeu també

El terme *intracode* l'expliquem en l'apartat següent, amb les imatges I.

Il·lustració 31. Exemple de GOP de $M = 9$ i $N = 3$ 

Ordre de visualització

Imatges I

La imatge I no té cap referència amb les altres, es comprimeix espacialment “a l’estil JPEG” (és a dir, es divideix en blocs de 8×8 píxels i es transforma, es quantifica, es codifica, etc.). Es codifiquen totes soles, i per això s’anomenen *intraframe*. La codificació utilitzada és la DCT bidimensional, semblant a la que hem vist en JPEG, i, com en JPEG, en cada imatge I es quantifiquen els coeficients, es disposen en sèrie en ziga-zaga i es codifiquen.

Les imatges I són les que ens apareixen en pantalla quan fem una cerca ràpida cap endavant o endarrere en el reproductor DVD o programari: un reproductor de baixa qualitat no necessita altres imatges, sinó només uns quants càlculs per a poder descomprimir la imatge I. És una manera econòmica de presentar-nos el punt aproximat de la pel·lícula on som.

Si codifiquéssim totes les imatges d’una seqüència com a imatges I, no tindríem una gran taxa de compressió i la pel·lícula no cabria en un DVD, perquè el fitxer seria de desenes de gigabytes. Cal buscar un mètode més agressiu de compressió per a la resta d’imatges, i tenim l’oportunitat de fer-ho, perquè l’ull humà no discerneix amb claredat objectes en moviment. Aquest mètode s’anomena *estimació de moviment*.

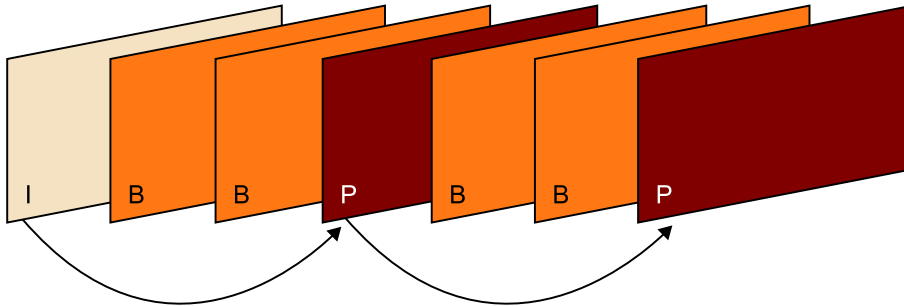
Imatges P i estimació de moviment

Tenint els “pilars” I d’una seqüència d’imatge, les imatges que hi ha entre dues imatges I es poden codificar a partir de l’estimació de moviment de dues maneres: imatges P i B.

Les **imatges P** (de **predictives**) es calculen a partir de la imatge I o la imatge P immediatament anterior (no vol dir que sigui l’anterior immediata, sinó l’última I o P que s’ha codificat).

També són de gran qualitat, encara que no tanta com la I, ja que es calculen mitjançant una estimació del moviment de la imatge (es divideix la imatge en macroblocs de 16×16 píxels i s'intenta veure quant s'han mogut en relació amb la imatge que prenen de referència, tant si és I com P).

Il·lustració 32



La intenció és que d'aquesta imatge en guardem només uns quants píxels i molts vectors de moviment que facin referència a quant s'han mogut els blocs de píxels de la imatge I o P de referència. Aquest procés proporciona un estalvi de bits impressionant. Aquí, doncs, el problema és calcular els vectors de moviment, és a dir, analitzar les diferències entre la imatge P i la de referència (I o P), comparar-les, veure si hi ha blocs de píxels semblants i detectar quant s'han desplaçat en horitzontal o vertical.

En seqüències estàtiques (la càmera quieta i els objectes que componen la imatge en repòs), les imatges P són nul·les. En seqüències en moviment, les imatges P augmenten en espai de memòria ocupat perquè han d'emmagatzemar el moviment dels píxels de la imatge de referència, i si a més apareixen objectes nous, els hi han d'afegir.

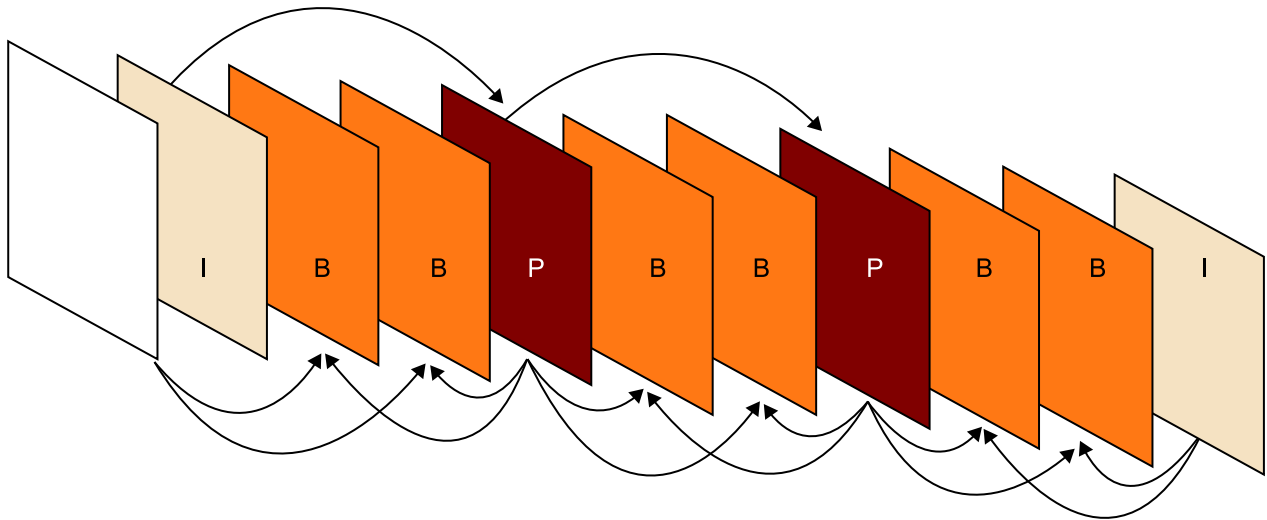
Imatges B

Les **imatges B** són el cas més extrem de compressió.

La **imatge B** (de **bidireccional predictiva**) es dedueix com a intermèdia a partir de dues imatges de tipus I o P, de manera que conté més informació d'estimació de moviment alhora que augmenta la compressió, encara que amb una qualitat relativament baixa.

Fixeu-vos que si bé les P usaven estimació de moviment en una sola direcció (des de la I o P anterior fins a elles), les B usen les dues direccions.

Il·lustració 33



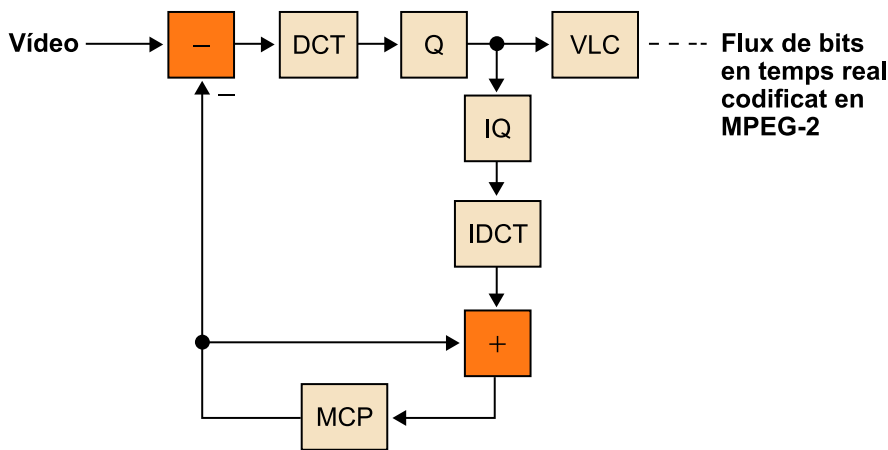
4.3.2. MPEG

Tot seguit explicarem en què consisteix la codificació MPEG, que es basa en els conceptes que acabem d'explicar sobre el GOP.

Estructura del codificador MPEG-2

Un diagrama que englobi els mecanismes que hem descrit pot ser el següent:

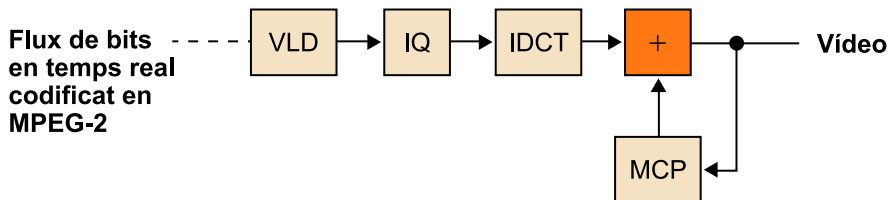
Il·lustració 34. Esquema de codificador MPEG-2



Al vídeo en temps real s'hi resta primerament la predicció que proposa el sistema i que genera pel llaç inferior de l'esquema, de manera que la DCT codifica la imatge "error de predicció". Després es quantifica (Q) i s'hi aplica codificació variable (VLC), i a més s'hi afegeix informació addicional (vectors del moviment, sincronització, etc.) per a ajudar el descodificador a fer la seva tasca. Per a fer la predicció, el flux de dades "error" quantificat passa per un desquantificador (IQ) i una DCT inversa (IDCT), i se suma a la predicció de compensació de moviment (MCP), que s'autoalimenta per reduir aquest error. Com més bo és el predictor de compensació de moviment, més bona és la resta

inicial i més reduït és el flux de dades error generat que finalment es transmet o s'emmagatzema, de manera que el canal pel qual es transmet o el dispositiu d'emmagatzematge pot ser que no necessitin tants requisits.

Il·lustració 35. Esquema d'un descodificador MPEG-2



El receptor descodifica la codificació variable (VLD), inverteix la quantificació (IQ) i, mitjançant una DCT inversa, recupera la imatge error, que se suma al resultat del predictor de compensació en moviment (MCP) per a donar la imatge final.

El control de la memòria intermèdia

Sens dubte, la sortida del codificador és un flux de dades variable, depenent del contingut visual del vídeo i de l'èxit del predictor de compensació de moviment, element clau d'aquest sistema.

No obstant això, en moltes ocasions aquest flux de dades s'ha de transmetre o emmagatzemar a una velocitat de bits fixa, de manera que s'afegeix una memòria intermèdia entre el codificador i el canal (i un altre entre el canal de recepció i el descodificador). Aquesta memòria intermèdia és omplerta de manera variable pel codificador i alimenta el canal de transmissió a velocitat constant, per la qual cosa cal prevenir tant que la memòria intermèdia quedi buida com que es desbordi. Per a això, la memòria intermèdia notifica al codificador el seu estat, i el codificador ajusta la velocitat de transmissió dels bits de sortida; per exemple, si la memòria intermèdia s'està omplint, el codificador augmenta la quantificació (Q) per reduir el volum de bits generats. Sens dubte, això afecta la qualitat del vídeo en recepció.

La inclusió d'una memòria intermèdia en emissió i una altra en recepció genera a més un retard, que es pot calcular com la capacitat utilitzada de la memòria intermèdia dividida entre la velocitat del canal. Aquest retard sol ser d'unes quantes desenes de segon.

MPEG-2 defineix la mida màxima que pot tenir aquesta memòria intermèdia, encara que el codificador és lliure d'utilitzar-ne tota la capacitat o només una part.

Propagació d'errors i compressió: l'etern dilema

Les imatges I poden contenir errors, tant propis de la codificació intraquadre a l'"estil JPEG" (una codificació amb pèrdues), com externs deguts a la transmissió (soroll, interferències, etc.) o a processos d'enregistrament no adequats (un suport DVD en males condicions, un làser de lectura esgotat, etc.). Ara bé, si tota la sèrie d'imatges P i B depenen d'aquelles imatges, és imaginable pensar que aquests errors de base es propagaran i les afectaran.

També les imatges P poden contenir errors, sia heretats de la I o P de què depenguin, sia per un error en la transmissió o l'enregistrament.

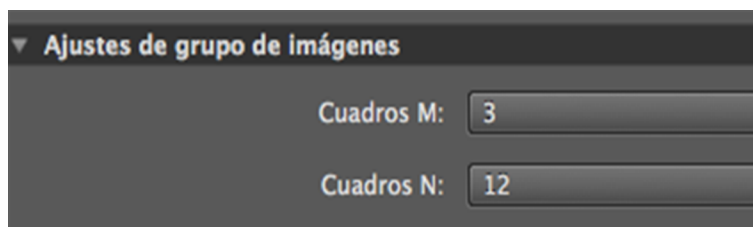
Les imatges B, senzillament, són les víctimes últimes de tots els errors de tota mena que s'hagin produït.

El valor N és la distància entre dues imatges I. Repassant la il·lustració 31, es dedueix que el GOP ha de tenir una mida de N imatges, ja que comença amb una imatge I i s'acaba just abans de la imatge I següent. Un valor de 12 és un equilibri entre una gran distància entre dues imatges de referència I i una compressió interessant amb una propagació d'errors acceptable.

El fet d'augmentar aquesta distància comportaria deixar en mans d'imatges P i B la qualitat de la seqüència i arriscar-se que petits errors de les imatges I s'amplifiquessin en tot el GOP. El fet de reduir-la seria millorar la qualitat però augmentar el pes de la seqüència i reduir-ne la compressió.

El valor M és la distància entre una imatge I i la següent imatge de tipus I o P que hi hagi. En l'exemple d'abans (il·lustració 31), és 3. Com més gran sigui el valor de M , més errors contindrà l'estimació de moviment però més alta serà la compressió: és l'etern dilema. En una exportació de projecte a MPEG-2 – per exemple amb un programari com Premiere, After Effects o Avid– deixa la tria a l'usuari:

Il·lustració 36. Exemple de selecció dels valors N i M en una autoria DVD amb MPEG-2



4.3.3. Exemples d'ús

Vegem alguns exemples d'ús habituals de la codificació MPEG-2:

- DVD: en fer una cerca d'imatge amb el comandament a distància en un DVD, el vídeo no flueix dinàmicament en pantalla, sinó fent salts. Això passa perquè va saltant d'imatge I a imatge I.
- En codificar una pel·lícula en MPEG-2, cada canvi de pla força l'inici d'un GOP. Així, la primera I conté els nous objectes.
- Reproducció en temps real per Internet: és habitual un GOP de $N = 12$ i $M = 3$.
- Edició de vídeo no lineal: com més imatges I hi hagi, millor. Fins i tot hi ha l'estàndard M-JPEG, que està format només per imatges I.

4.4. Els fluxos elementals de dades

Una vegada generades les imatges I, P i B, a partir d'aquestes cal construir un flux final de bits per a emmagatzemar o transmetre, i aquest procés té diverses etapes. Comencem per definir el *flux elemental* (ES²¹), que és senzillament la sortida del codificador, que conté tota la informació necessària perquè un descodificador pugui crear una aproximació tan precisa com sigui possible al vídeo original.

⁽²¹⁾De l'anglès *elementary stream*.

Hi ha dos tipus diferents d'ES: els d'àudio (AES²²) i els de vídeo (VES²³). Tot seguit descriurem cadascun d'aquests ES.

⁽²²⁾De l'anglès *audio elementary stream*.

4.4.1. Flux elemental de vídeo

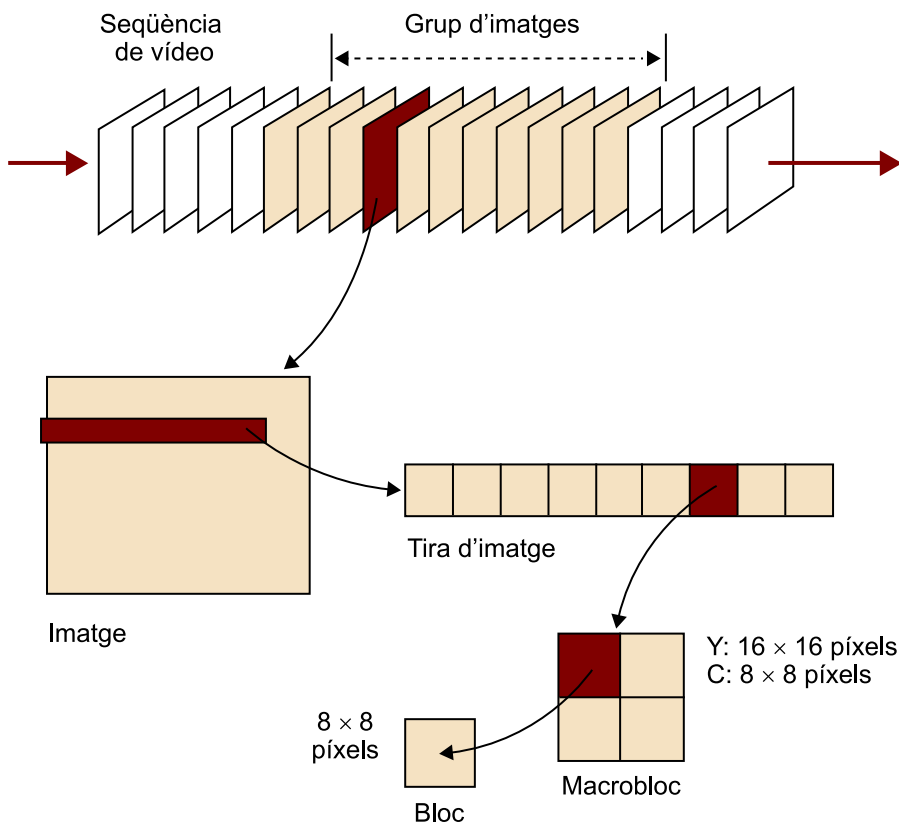
⁽²³⁾De l'anglès *video elementary stream*.

Un **flux elemental de vídeo (VES)** és una seqüència de vídeo. En general la sortida del codificador ja és en format VES.

⁽²⁴⁾En anglès, *player*.

Com es veu en la il·lustració 37, el flux no conté cada imatge per separat, sinó que s'estructura en grups d'imatges (GOP), la unitat de treball de qualsevol reproductor que vulgui presentar aquest flux (un reproductor²⁴ DVD, un programa de reproducció, etc.):

Il·lustració 37



Construcció del VES

Cada imatge es converteix en tres matrius rectangulars de valors, cadascuna amb els valors de Y , Cb i Cr , respectivament. En general, la matriu I és quatre vegades més gran que les altres dues, perquè com hem vist és més rellevant la luminància (Y) que no pas els components de color (Cb , Cr).

En funció del contingut i de la comparativa que fa entre imatges consecutives, el codificador pren la decisió de definir una de les imatges com de tipus I , és a dir, codificable de manera independent i pilar bàsic per a la codificació de les immediates. Acostuma a passar que, per exemple, la primera imatge després d'un canvi de pla es defineixi com a I , ja que no té relació amb les anteriors, però el criteri és molt més complex i cada fabricant aplica el que considera més convenient (l'algorisme no està definit en l'estàndard).

A partir de la imatge I , es calculen les imatges P posteriors i les B tal com hem explicat. Cal destacar que dins un GOP sempre es manté la mateixa seqüència d'imatges una vegada definits els paràmetres M i N de la codificació.

És important recalcar que MPEG no especifica en els estàndards com ha de ser un codificador, sinó que especifica la sortida del codificador, per a assegurar-se així de la compatibilitat amb el descodificador. D'aquesta manera, els codi-

Vegeu també

Les imatges I , P i B s'expliquen en el subapartat 4.3.1 d'aquest mòdul didàctic.

ficadors i descodificadors de diferents fabricants poden interoperar, i a més permeten a la indústria investigar i millorar les seves tècniques de codificació continuant dins l'estàndard.

L'estàndard, en canvi, sí que especifica un mecanisme de descodificació (un descodificador), però aquest mecanisme s'ha de prendre com a referència, i també s'ha de deixar llibertat a la indústria perquè el canviï sempre que asso-leixi el mateix resultat que l'estàndard (en aquest sentit es pensa en millores d'execució i adaptació del mètode a un maquinari específic).

En MPEG-2 hi ha moltes implementacions de codificadors i descodifi-cadors, algunes per a maquinari i moltes per a programari. La qualitat de codificació MPEG-2 ha millorat sensiblement des que va sortir i la qualitat és alta fins i tot amb fluxos de menys de 2 Mbps. És un estàndard madur, però amenaçat per MPEG-4, més actualitzat, que el supera en tots els aspectes, excepte en l'econòmic.

Les solucions d'IPTV solen utilitzar equipament dedicat i costós per a fer la codificació. La descodificació es fa en l'equip del client, en general en progra-mari (navegadors d'Internet), però hi ha algunes excepcions en maquinari (te-levisió per cable).

4.4.2. Flux elemental d'àudio

Un **flux elemental d'àudio (AES)** és la sortida del codificador d'àudio.

Aquest cas és molt diferent del de compressió de vídeo, perquè aquí no hi ha diferents tipus de quadres, sinó que es codifiquen tots de la mateixa manera, per la qual cosa tots tenen la mateixa mida.

4.4.3. Flux elemental empaquetat

A l'hora d'agrupar l'*stream* de vídeo (VES) i el d'àudio (AES) en un sol flux de dades, cal anar amb compte de no generar retards entre l'un i l'altre (el fet d'enviar massa vídeo i poc àudio implicaria no tenir prou àudio o tenir-ne de retardat en relació amb el vídeo). Cal dividir els dos fluxos de dades en paquets d'una mida adequada abans de posar-los seqüencialment l'un darrere l'altre.

Aquest procés s'anomena **empaquetatge** i implica indicar en cada paquet el tipus a què pertany i uns identificadors perquè el receptor els pugui reconstruir correctament, i també dades addicionals per a assegurar la sincronització (quin fragment d'àudio va amb quin fragment de vídeo).

El flux total és el flux elemental empaquetat (PES).

4.5. Emmagatzematge i transmissió

En general, cal combinar diversos PES (almenys un àudio i un vídeo) per a crear un contingut multimèdia que després serà reproduït. Llavors sorgeixen dues possibilitats de tractament: emmagatzemar el contingut per a reproduir-lo o transmetre'l després (en el nostre cas, per una xarxa IP).

4.5.1. Emmagatzematge

L'estàndard d'MPEG-2 no especifica cap format de contenidor d'arxiu. Això també passa amb MPEG-1, H.263 i d'altres, de manera que es poden fer servir molts contenidors: MP3 per a àudio, MOV per a vídeo, etc.

4.5.2. Transmissió

L'estàndard del sistema d'MPEG-2²⁵ defineix dos mètodes de combinació de les dades d'àudio, vídeo i contingut associat per a la transmissió en un únic flux de dades:

- **Program Stream** (conegut com a *MPEG-2 PS* o simplement *MPEG-PS*).
- **Transport Stream** (conegut com a *MPEG-2 TS* o simplement *MPEG-TS*).

Un canal (com un canal de TDT) està format almenys per un PES de vídeo i un PES d'àudio. Del procés d'ajuntar uns quants PES que es reproduiran conjuntament, se'n diu *multiplexació*. Per a aconseguir una reproducció conjunta sincronitzada (sense desfasament entre àudio i vídeo), cal enviar informació de rellotge en la multiplexació.

Mentre que MPEG-PS permet transportar un únic canal, MPEG-TS permet enviar-ne uns quants alhora (cadascun amb una sincronització de rellotge independent). MPEG-TS incorpora, a més, mecanismes de detecció i correcció de fallades en la transmissió, tan habituals en xarxes públiques com Internet o 3G. Ho veurem al final del mòdul per la rellevància que té.

Respecte a la transmissió per xarxa, es defineix un conjunt de protocols per a la transmissió de contingut multimèdia per una xarxa IP. Entre aquests protocols cal destacar, per a sistemes de vídeo digital de temps real, el protocol de transmissió en temps real (*real-time transport protocol*, RTP).

⁽²⁵⁾ISO/IEC 13818-1.

Vegeu també

Els mecanismes de detecció i correcció de fallades en la transmissió s'estudien en l'apartat 5 d'aquest mòdul didàctic.

Vegeu també

Els protocols per a la transmissió de contingut multimèdia per una xarxa IP s'estudien en el subapartat 5.4 d'aquest mòdul didàctic.

4.6. Perfils i nivells en MPEG-2

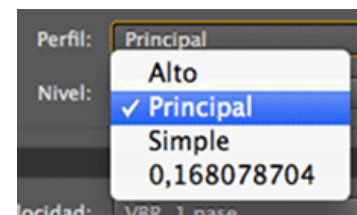
MPEG-2 es va dissenyar per a incloure la màxima quantitat de sistemes de vídeo digital presents en el moment de configuració de l'estàndard. Això implica disposar de diferents graus de qualitat per a cada aplicació, en general determinats per la taxa de bits i la resolució de la codificació.

Els graus de qualitat estan definits en l'estàndard en **perfils (profiles)** i **nivells (levels)**.

4.6.1. Perfils

El perfil defineix la resolució de l'espai de colors i l'escalabilitat del flux de bits. Les opcions més habituals són les següents:

- **Perfil alt (high profile, HP):** destinat a codificar senyals d'alta definició. En realitat, no es fa servir mai ja que en aquest cas l'MPEG-4 és molt més efectiu.
- **Perfil principal (main profile, MP):** és el més habitual, adreçat a assolir una qualitat estàndard de vídeo. Activa l'ús d'imatges I, P i B en la codificació, i possibilita codificar senyals de vídeo entrellaçats.
- **Perfil simple (simple profile, SP):** necessari quan el flux de dades generat va adreçat a equips de baixes prestacions tècniques –telefonía mòbil, PDA, tauletes, etc.–. No fa servir imatges de tipus B, cosa que simplifica el càlcul de la descodificació i la necessitat de memòries intermèdies.



Il·lustració 38. Perfils d'exportació (Adobe Premiere)

4.6.2. Nivells

El nivell defineix la resolució d'imatge, la quantitat de mostres de luminància (Y) per segon, la quantitat de capes d'àudio i vídeo i el màxim de la taxa de bits per perfil:

- **Nivell alt (high level, HL):** 1.920×1.152 , 80 Mbps. Per a alta definició no s'ha utilitzat mai.
- **Nivell alt 1440 (high-1440, H-14):** 1.440×1.152 , 60 Mbps. Per a alta definició tampoc no s'ha utilitzat mai.
- **Nivell principal (main level, ML):** 720×576 , 15 Mbps. L'habitual per a imatges de qualitat estàndard.



Il·lustració 39. Nivells d'exportació (Adobe Premiere)

- **Nivell baix (low level, LL):** 288 × 352, 4 Mbps.

4.6.3. Combinacions de perfils i nivells

No es fan servir totes les combinacions de perfils i nivells. La notació utilitzada és perfil@nivell. Les més habituals són les següents:

| Perfil@nivell | Resolució | Hz | Mostratge | bps | Exemple d'aplicació |
|---------------|---------------|-------------|-----------|---------------|--|
| SP@LL | 176 × 144 | 15 | 4:2:0 | 0,096 | Tauletes, mòbils |
| SP@ML | 352 × 288 | 15 | 4:2:0 | 0,384 | PDA |
| | | 320 × 240 | 24 | | |
| MP@LL | 352 × 288 | 30 | 4:2:0 | 4 | Descodificadors |
| MP@ML | 720 × 480 | 30 | 4:2:0 | 15 (DVD: 9.8) | DVD, TDT, TV per cable i per satèl·lit |
| | | 720 × 576 | 25 | | |
| 422P@ML | 720 × 480 | 30 | 4:2:2 | 50 | Sony IMX |
| | | 720 × 576 | 25 | | |
| 422P@H-14 | 1.440 × 1.080 | 30 | 4:2:2 | 80 | Reservat |
| | | 1.280 × 720 | 60 | | |
| 422P@HL | 1.920 × 1.080 | 30 | 4:2:2 | 300 | Reservat |
| | | 1.280 × 720 | 60 | | |

4.7. MPEG-2 en la indústria

Avui dia MPEG-2 és molt utilitzat tant en emmagatzematge com en transmissió d'àudio i vídeo. Alguns dels sistemes que fan servir MPEG-2 són DVD i els estàndards de TDT, satèl·lit i cable a Europa (DVB) i Amèrica (ATSC). En cada cas només s'implementa la part necessària de l'estàndard.

Vegem uns exemples reals merament il·lustratius d'MPEG-2 per a emmagatzematge:

a) DVD²⁶

⁽²⁶⁾De digital video disc.

- Contenedor de vídeo: MPEG-2 Program Stream.
- Codificació de vídeo: MPEG-2.
- Codificació d'àudio: PCM, MP2 (Musicam), Dolby Digital o DTS a Europa, i Dolby Digital o DTS als Estats Units.

- Resolució (píxels):
 - NTSC: 720 × 480, 704 × 480, 352 × 480, 352 × 240
 - PAL: 720 × 576, 704 × 576, 352 × 576, 352 × 288
- Taxa de bits d'“àudio + vídeo”: pic 15 Mbps, mínim 300 kbps.
- Relació d'aspecte: 4:3, 16:9, 2,21:1.
- Taxa de quadres: 29,97 quadres per segon (NTSC), 25 quadres per segon (PAL).
- YUV 4:2:0.

b) DVB²⁷

⁽²⁷⁾De digital video broadcast.

Inclou la televisió digital terrestre (DVB-T), per satèl·lit (DVB-S) i per cable (DVB-C). Resolucions possibles:

- SDTV:
 - 720, 640, 544, 480 o 352 × 480 píxels, 24/1.001, 24, 30/1.001 o 30 quadres per segon.
 - 720, 704, 544, 480 o 352 × 576 píxels, 25 quadres per segon.
 - 352 × 240 píxels, 24/1.001, 24, 30/1.001 o 30 quadres per segon.
 - 352 × 288 píxels, 25 quadres per segon.

4.8. Mesura de la qualitat d'imatge en MPEG-2 per a DVB

Tal com hem vist per a imatge fixa, hi ha mètodes per a analitzar la qualitat de vídeo comprimit en MPEG-2. En DVB hi ha dues grans fonts de problemes:

1) Els generats pel soroll en el canal de transmissió. Mentre que en la televisió analògica el soroll es plasmava com una coneguda neu, en la TDT genera un augment del nombre d'errors de bit que fins a cert punt són recuperats pels sistemes de protecció d'errors que té i que no són visibles ni audibles. A banda d'aquest punt, poden aparèixer blocs d'imatge incorrectes, efecte mosaic en la imatge i, en el cas extrem, el servei cau completament i la pantalla es torna negra. Aquests fenòmens són familiars en TDT (DVB-T) i satèl·lit (DVB-S), però no ho són gaire en cable (DVB-C) perquè és un canal més controlat i fiable.

2) Els artefactes produïts per la mateixa codificació MPEG-2. Centrant-nos en aquesta segona font de problemes, la codificació MPEG-2 té un efecte directe en la qualitat d'imatge pel nivell de severitat en la compressió utilitzada. En dividir-se la imatge en macroblocs de 16 × 16 píxels de manera independent als blocs circumdants, quan s'aplica una alta compressió apareixen discontinuïtats de luminància i crominància entre blocs adjacents, perceptibles per

l'ull. D'això se'n diu *blocking*. A més, aquesta unitat bàsica de macrobloc és la base de la codificació predictiva entre quadres I, i també la base de generació dels quadres P, de manera que el problema és intrínsec a l'estàndard.

Sens dubte, el fenomen no depèn només del nivell de compressió, sinó també del contingut visual de la imatge. Una situació clàssicament complexa són els esports: la Fórmula 1 és molt més difícil de comprimir sense artefactes que no pas una partida d'escacs pel moviment ràpid i detall de la primera.

Hi ha un sistema estandarditzat de mesura subjectiva de la qualitat anomenat ITU-R BT.500 que descriu els mètodes d'anàlisi de la qualitat del vídeo a partir dels resultats donats per un grup de persones seleccionades com a observadores.

Hi ha dos mètodes subjectius principals:






- L'escala de qualitat contínua de doble estímul (DSCQS): un grup de persones compara un clip de vídeo original amb la versió codificada d'aquest clip, i la valoren en un rang de 0 (dolent) a 100 (excel·lent).
- L'escala de qualitat contínua d'estímul únic (SSCQE): un grup de persones valora el clip codificat, de 0 a 100, sense veure el clip original.

No hi ha mètodes estàndards per a l'anàlisi objectiva de la qualitat MPEG-2. Cada fabricant d'equips d'instrumentació i mesura han implementat els seus, però en general se centren en la comparació dels píxels que voregen cada macrobloc, responsables finals de l'efecte de *blocking*. No fan una comparativa directa (com es podria fer amb una imatge estàtica), sinó que tenen en compte el que anomenen *activitat espacial* (riquesa d'altres freqüències espacials) i *activitat temporal* (moviment), perquè tant l'una com l'altra podrien falsejar els resultats de les mesures.



Il·lustració 40. Equip Rohde&Schwarz, analitzador de qualitat de vídeo digital

Il·lustració 41. Resultats de l'anàlisi de qualitat de cinc clips de vídeo amb l'analitzador DekTec DTC-352

| Video Quality Analysis | | | | | | | | |
|---|--------------|-----------|---|--------------------------------------|-----------------|----|----|----------|
| Thumbnail | Filename | File Type | File Summary | Size / Duration | Average Quality | | | Status |
| | | VIDEO | | | BAD | 40 | 60 | |
|  | Demo.mpeg | VIDEO | MPEG-2 video 720 x 576 9000000 B/s 25 F/s | Filesize : 32 MB Duration: 30 sec | 84 | | | analysed |
|  | cities.h264 | VIDEO | H.264 / AVC 704 x 576 25 F/s | Filesize : 12 MB | 32 | | | analysed |
|  | KMBC_TV.mpeg | VIDEO | MPEG-2 video 720 x 480 3250000 B/s 29.97 F/s | Filesize : 55 MB Duration: 1 min | 12 | | | analysed |
|  | ARD.ts | VIDEO | MPEG-2 video 544 x 576 7000000 B/s 25 F/s | Filesize : 83 MB Duration: 52 sec | 83 | | | analysed |
|  | KTVI_TV.mpeg | VIDEO | MPEG-2 video 720 x 480 3250000 B/s 29.97 F/s | Filesize : 55 MB Duration: 1 min | 9 | | | analysed |

Com que l'estàndard MPEG-2 no defineix el procediment de codificació, en els darrers anys hi ha hagut millores substancials de diversos fabricants que han posat la qualitat d'MPEG-2 molt per sobre dels primers codificadors que hi havia, en un afany també de no perdre terreny davant l'auge del seu successor, l'MPEG-4, que veurem amb detall en l'apartat següent.

5. L'estàndard MPEG-4 i H.264/AVC

Després de l'MPEG-2, que es va acabar el 1995, el grup MPEG continua la seva feina d'estandardització. El 1993 (abans d'haver acabat l'MPEG-2) comencen la feina en l'estàndard MPEG-4, que inclou molts més aspectes dels sistemes de vídeo digital que no pas els predecessors MPEG-1 i MPEG-2.

La part 2 de l'estàndard MPEG-4 (ISO/IEC 14496-2), coneguda com a *MPEG-4 Visual*, especifica la codificació i descodificació de vídeo digital. MPEG-4 Visual es va acabar d'estandarditzar el 1999. L'estàndard H.264 el va començar el grup de treball Video Coding Experts Group (VCEG²⁸) de la Unió Internacional de Telecomunicacions (UIT-T). Les darreres etapes del treball les va fer la Joint Video Team (JVT), un grup conformat pels grups VCEG i MPEG.

⁽²⁸⁾VCEG és la sigla de *Video Coding Experts Group*.

L'estàndard final es va publicar en conjunt el 2003. Per tant, s'ha d'entendre que MPEG-4 i H.264 no són sinònims: H.264 és una part de l'estàndard MPEG-4 i també és conegut com a *H.264 Advanced Video Coding (AVC)*. Per tant, les formes correctes d'anomenar aquest estàndard són **MPEG-4 Part 10** o **H.264/AVC**.



MPEG-4 Visual i H.264, encara que contemporanis, són molt diferents de concepció: MPEG-4 Visual apunta a la flexibilitat i mira d'incloure la màxima quantitat possible de sistemes de vídeo digitals, mentre que H.264 apunta a l'eficàcia en la compressió i la confiança de la transmissió.

La publicació de l'estàndard el 2003

L'estàndard final va ser publicat per l'MPEG com la part 10 de l'estàndard MPEG-4 (ISO/IEC 14496-10), i la UIT-T com a H.264.

La codificació utilitzada en la majoria dels sistemes d'IPTV és H.264, i per això dedicarem aquest apartat només a aquesta part de l'estàndard MPEG-4. Per a entendre'l més bé, l'estructura d'aquest apartat és molt semblant a la de l'MPEG-2, de manera que començarem pel detall dels estàndards que la componen, continuarem amb uns quants detalls del format, els mecanismes de transmissió i emmagatzematge, els perfils i nivells definits, i acabarem amb l'aplicació actual d'aquesta codificació en la indústria.

El contenidor d'aquest còdec està ben especificat i es basa en un format estàndard que curiosament es va inspirar en el de QuickTime d'Apple: l'ISO Media File Format. L'extensió dels arxius MPEG-4 és .mp4.

5.1. L'estàndard i les patents

MPEG-4 està estandarditzat en la norma ISO/IEC 14496. Igual que per a MPEG-2, es compon de diverses parts, entre les quals hi ha les següents:

- El sistema: descriu la sincronització i la transmissió simultània d'àudio i vídeo.
- El vídeo: el còdec de compressió per a elements visuals (vídeo, textures, imatges sintètiques, etc.). Un dels molts perfils definits en la part 2 és l'Advanced Simple Profile (ASP).
- L'àudio: el conjunt de còdecs de compressió per a la codificació de fluxos d'àudio; inclouen variants d'Advanced Audio Coding (AAC) i eines de codificació d'àudio i parla.
- Transport sobre xarxes IP: especifica un mètode per a transportar contingut MPEG-4 sobre xarxes IP.
- H.264 (Advanced Video Coding, AVC): un còdec estandarditzat de senyals de vídeo.
- Enginyeria d'aplicació i descripció d'escenes (BIFS). Per a contingut interactiu 2D i 3D.
- Format per a mitjans audiovisuals basat en ISO: un format d'arxius per a emmagatzemar contingut multimèdia.
- Extensions per al maneig i la protecció de propietat intel·lectual (IPMP).
- El format d'arxiu contenidor designat per a continguts MPEG-4.
- El format d'arxiu AVC per a l'emmagatzematge de vídeo.

És important destacar la part 10 de l'estàndard que especifica l'H.264 i la part 8 que especifica el transport d'MPEG-4 sobre una xarxa IP.

5.2. H.264 o MPEG-4 Part10

En la seva filosofia, la codificació H.264 no difereix gaire d'MPEG-2. S'especifica el format de codificació i com s'ha de fer per a descodificar-lo (i es torna a deixar lliure a la indústria el disseny de codificadors eficients).

També fa servir la nomenclatura de *perfils* i *nivells* per a definir variants de resolució i qualitat de vídeo. Avui dia no hi ha gaires perfils en l'estàndard que estiguin definits per complet. Al començament només es van definir tres perfils: *baseline*, *main* i *extended*. Actualment, però, sembla que es perfila amb més acceptació en el mercat el perfil *high*, com veurem més endavant.

Una millora en relació amb l'estàndard MPEG-2 és que se separa la codificació de la transferència, cosa que en facilita la implementació.

El format de sortida del codificador s'anomena *Video Coding Layer* (VCL), que bàsicament és una seqüència de bits que representa vídeo codificat. La jerarquia de codificació de vídeo MPEG-2 (GOP, quadres, blocs) queda gairebé inalterada.

Un problema habitual en MPEG-2 és que si hi ha diferents subfluxos o serveis continguts en el mateix flux de bits, en commutar d'un a un altre, el descodificador perdia qualitat un instant. Contra això, en els perfils més alts d'H.264 apareix la possibilitat d'utilitzar dos tipus extrems de quadres (a més dels I, P, B):

- **SP (*switching P*)**, per a facilitar canviar de flux de codificació (conté macroblocs I o P).
- **SI (*switching I*)**, per a facilitar canviar de flux de codificació (conté macroblocs SI).

5.3. Emmagatzematge

L'estàndard MPEG-4 configura una especificació d'emmagatzematge en disc, com hem comentat al principi d'aquest apartat. Es basa en l'estàndard ISO Media File Format, que està inspirat en el format QuickTime d'Apple. L'extensió dels arxius MPEG-4 és .mp4.

Hi ha molts més formats contenidors que tenen aquesta codificació, com, per exemple, 3GPP, MOV o Flash Video.

5.4. Transmissió

Tant MPEG-4 com H.264 no defineixen un sistema específic de transport per a la transmissió.

Amb l'esmena 3 del sistema MPEG-2 (ISO/IEC 13818-1), es pot transportar MPEG-4, i en particular H.264, de la manera següent:

- **Program Stream** (conegut com a *MPEG-2 PS* o simplement *MPEG-PS*).
- **Transport Stream** (conegut com a *MPEG-2 TS* o simplement *MPEG-TS*), que permet fer conviure canals TDT en H.264/AVC en HD amb els MPEG-2 en SD.

Respecte a la transmissió per xarxa IP, s'ha actualitzat el protocol de temps real RTP per a la transmissió d'MPEG-4 i H.264. És relativament senzill transmetre contingut multimèdia en una xarxa de paquets IP; fins i tot limitant-nos al context d'IPTV, no es fa servir només RTP. En les xarxes actuals, per a transmissió de vídeo digital per Internet hi ha una clara preferència per protocols basats en TCP, i en particular, per als que es basen en HTTP, per la seva compatibilitat amb les configuracions freqüents de tallafocs²⁹ i servidors intermediaris³⁰.

⁽²⁹⁾En anglès, *firewalls*.

⁽³⁰⁾En anglès, *proxies*.

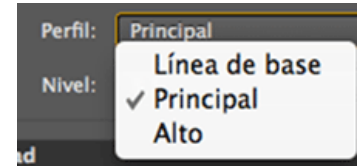
5.5. Perfils i nivells

Com en l'MPEG-2, l'estàndard H.264/AVC té diferents graus de qualitat, que en l'estàndard estan definits en termes de perfils i nivells.

5.5.1. Perfils

En l'estàndard hi ha set perfils, i ens centrarem en els tres més habituals:

- **Línia de base** (*baseline profile, BP*): usat per a dispositius senzills com ara reproductors portàtils i telefonia mòbil, i també per a reproducció en temps real de vídeo per xarxa IP.
- **Principal** (*main profile, MP*): adreçat a *broadcast* (TDT HD) i emmagatzematge (Blu-ray), proveeix d'alta definició.
- **Alt** (*high profile, HP*): per a aplicacions professionals que requereixen alta definició de crominància i luminància (4:4:4).



5.5.2. Nivells

Representats amb un nombre de l'1.0 al 5.1, són configuracions per a velocitats de transmissió creixents (des de 64 kbps en 1.0 fins a 240 Mbps en 5.1).

5.6. H.264 en la indústria

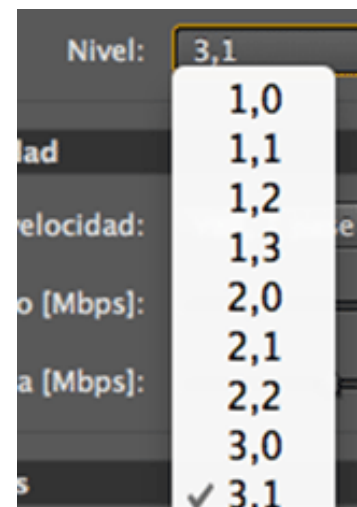
L'H.264 sempre és el triat en sistemes de vídeo digital d'alta resolució, com les xarxes d'IPTV i la televisió en HD terrestre, per cable o per satèl·lit. És d'esperar que amb el pas del temps els diferents sistemes de transmissió de vídeo migraran a aquest format i reemplaçaran així els tradicionals H.262/MPEG-2, tal com ha ocorregut a França, que ha apostat des del començament de les emissions TDT per aquest format.

Alguns dels sistemes que utilitzen o plantegen utilitzar H.264 són aquests:

a) Per a emmagatzematge:

Blu-ray Disc de Blu-ray Disc Association (BDA). Fa servir un perfil alt (HP). Com a característiques més conegudes de Blu-ray tenim aquestes:

- **Contenedor de vídeo:** BDAV MPEG-2 Transport Stream (fitxers d'extensió .bdav), de manera que l'enregistrament d'una emissió de TDT en HD es pot emmagatzemar directament sense alterar en absolut el format.



L'HD-DVD de Toshiba

El desaparegut format d'emmagatzematge HD-DVD de Toshiba tenia característiques semblants.

- Codificació de vídeo: H.264/MPEG-4 AVC (o VC-1 de Microsoft).
- Codificació d'àudio: Dolby Digital, DTS, Dolby Digital Plus, DTS-HD, PCM.
- Combinacions de resolució i taxa de quadres en modes progressius:
 - **1080p**: 1.920 × 1.080 píxels a 24/23,97 quadres per segon (Estats Units / Europa).
 - **720p**: 1.280 × 720 píxels a 50/59,94 quadres per segon (Estats Units / Europa).
 - I altres combinacions progressives (1.440 × 1.080, 1.280 × 720) a 24 quadres per segon (mode cinema).
- Combinacions de resolució i taxa de quadres en modes entrellaçats:
 - **1080i**: 1.920 × 1.080 píxels a 25/29,97 quadres per segon (Estats Units / Europa), o 1.440 × 1.080 píxels a 25/29,97 quadres per segon (Estats Units / Europa).
 - **720i**: 1.280 × 720 píxels a 25/29,97 quadres per segon (Estats Units / Europa).
- Relació d'aspecte 16:9 (encara que hi ha dos modes 4:3 no gaire utilitzats).
- Taxa de bits d'"àudio + vídeo": pic 36 Mbps.
- YUV 4:2:0.

b) Per a *broadcast*:

- El consorci europeu DVB va apostar des del 2004 per aportar (si no reemplaçar-la) a la codificació H.262/MPEG-2 actual aquest nou estàndard, tant en terrestre (DVB-T) com per satèl·lit (DVB-S) com per cable (DVB-C). El nou estàndard DVB-T2, que s'implantarà a partir del 2015 a Europa, optimitzarà l'aprofitament de les freqüències de canal per a donar entrada a més canals, cosa que podria facilitar la migració a H.264.
- El comitè nord-americà ATSC especifica H.264 i VC-1 com a estàndards per a la transmissió terrestre de televisió. L'associació japonesa ARIB també incorpora H.264/AVC al seu sistema de TDT (ISDB-T).
- Per satèl·lit és present en la majoria de plataformes (Digital+, BBC HD, Euro1080, etc.).

c) Per a xarxa de telefonia mòbil analògica:

- The 3rd Generation Partnership Project (3GPP) inclou H.264/AVC com a opcional. Avui dia (2011) no és gaire freqüent poder rebre aquest servei per l'amplada de banda que fa falta.

5.7. Gestió d'objectes multimèdia amb MPEG-4

Fins ara, MPEG-4 es pot considerar una versió actualitzada d'MPEG-2, però és més que un sistema de codificació audiovisual: proveeix d'eines per a representar "objectes *media*", és a dir, unitats visuals o auditives que poden ser sintètiques o capturades de la realitat.

MPEG-4 possibilita descriure aquests **objectes multimèdia** i crear-hi "composicions", és a dir, escenes, i fins i tot proveeix d'estàndards per a la "interacció" de l'usuari amb aquestes escenes audiovisuals.

MPEG-4 possibilita produir continguts als autors, i així potencia la reutilització d'una manera més gran que no pas la que s'aplica avui dia en televisió digital, web, etc. Per als proveïdors de serveis de xarxa, ofereix informació transparent que pot ajudar a millorar el transport d'extrem a extrem per xarxes heterogènies. Per acabar, per als usuaris finals ofereix interacció i multimèdia fins i tot a velocitats de transmissió de bits baixes, com les comunicacions mòbils.

No és la faceta més coneguda de l'estàndard, però sens dubte és la que encara té més camí de desenvolupament i evolució.