

5. TEHNICI DE WATERMARKING PENTRU VIDEO

În acest capitol vom vorbi despre metode de watermarking specifice pentru video. Secvențele video sunt formate dintr-o serie de imagini consecutive, egal depărtate în timp. Astfel, problema generală a watermarking-ului pare foarte asemănătoare pentru imagini și secvențe video și ideea că tehnicile watermarking pentru imagini pot fi aplicate pentru secvențe video este evidentă. Acest lucru este parțial adevărat și există destul de multe publicații care se încheie cu observația că abordarea propusă este aplicabilă și pentru secvențe video. Oricum, există diferențe importante între imagini și video, ce sugerează o abordare specifică pentru video.

O diferență importantă este spațiul disponibil al semnalului. Pentru imagini, spațiul semnalului este foarte limitat. Acest lucru motivează mulți cercetători să folosească modele implicite sau explicite ale Sistemului Vizual Uman pentru a atinge granița invizibilității și a insera un watermark cât se poate de robust fără a sacrifica calitatea imaginii.

Pentru secvențe video spațiul disponibil al semnalului este mult mai larg. Pe de altă parte, watermarking-ul pentru video impune constrângeri de timp real sau aproape timp real asupra sistemului. Ca o consecință a acestui lucru, folosirea metodelor de watermarking bazate pe modele explicite ale SVU este mai puțin importantă și pentru multe aplicații chiar prohibitivă din punct de vedere al complexității. Gradul de complexitate este, în general, o caracteristică mult mai importantă pentru aplicațiile de watermarking video decât pentru aplicațiile pentru imagini.

Pentru aplicații specifice, de exemplu amprentarea secvențelor video (inserarea unui identificador de destinatar), această problemă este chiar mai gravă, deoarece secvențele video sunt de obicei stocate în format comprimat. În comparație cu imaginile, stocarea în format necomprimat și compresia sau decompresia în timp real, inserarea watermark-ului și

recomprimarea, nu sunt de obicei fezabile pentru astfel de aplicații. Astfel, acest tip de aplicații necesită inserarea watermark-ului în domeniul comprimat.

O altă observație de luat în calcul este faptul că structura unei secvențe video văzută ca succesiune de imagini creează problema atacurilor specifice, ca de exemplu medierea de cadre, eliminarea de cadre și interschimbarea de cadre [152, 153]. La o frecvență a cadrelor de 25-30 Hz, ca cea folosită în televiziune, aceste atacuri nu ar putea fi percepute de un observator normal. Dar o schemă bună de watermarking ar trebui, totuși, să reziste acestor tipuri de atacuri, de exemplu prin distribuirea informației watermark în mai multe cadre consecutive. Pe de altă parte, ar fi de preferat ca întreaga informație inserată să poată fi extrasă dintr-o mică parte a secvenței video. Compromisul între aceste două cerințe contradictorii se face în funcție de aplicație.

Dacă în domeniul watermarking-ului pentru imagini au fost publicate foarte multe lucrări, există mai puține publicații care să se ocupe cu watermarking-ul video digital. Totuși, interesul pentru astfel de tehnici e mare, de exemplu, pentru standardul DVD se dorește un sistem eficient de protecție la copiere bazat pe watermarking. Scopul este de a marca orice material video astfel încât toate player-ele și recorder-ele să refuze rularea sau înregistrarea materialelor piratate.

5.1. Provocări în domeniul watermarking-ului video

Cu toate că mulți algoritmi de watermarking care au fost inițial dezvoltați pentru imagini pot fi folosiți pentru inserarea și extragerea informației watermark în/din secvențe video brute, există câteva provocări de bază în domeniul watermarking-ului video ce trebuie adresate și care sunt specifice acestui tip de conținut.

O primă provocare de interes specific pentru comunitatea de watermarking video sunt distorsiunile video neagresive ce pot distruge informația ascunsă sau capacitatea de detecție a watermark-ului. O a doua provocare sunt atacurile deliberate asupra sistemului de watermarking. Robustețea watermark-ului digital a fost întotdeauna evaluată prin integritatea watermark-ului după distorsiuni/atacuri. Au fost dezvoltate chiar și unelte speciale de evaluare pentru automatizarea acestui proces [154]. Distorsiunile și atacurile asupra sistemelor de watermarking sunt prezentate pe larg în Capitolul 6 și nu vor mai fi discutate și aici. În subcapitolul următor ne vom concentra, în schimb, pe o altă provocare importantă

pentru sistemele de watermarking video, și anume, procesarea în timp real care este destul de multe ori o cerință pentru prelucrarea video.

5.2. Watermarking în timp real

Procesarea în timp real ar putea fi o specificație în plus pentru watermarking-ul video. Pentru imagini acest lucru nu a fost o grijă reală. Dacă cineva dorește să insereze un watermark sau să verifice prezența unui watermark într-o imagine, o durată de câteva secunde este o întârziere acceptabilă. Dar o astfel de întârziere este nerealistă în contextul video. Cadrele sunt trimise cu o frecvență destul de mare, de obicei 25-30 de cadre pe secundă, pentru a se obține o mișcare continuă. Cel puțin codorul sau detectorul de watermark, în unele cazuri chiar ambele, ar trebui să producă o astfel de rată a cadrelor. În contextul monitorizării emisiei, detectorul ar trebui să fie în stare să detecteze un watermark în timp real. Într-un mediu VOD (Video On Demand – video la cerere) server-ul video ar trebui să poată insera watermark-ul, ce îl identifică pe utilizator, la rata cadrelor cu care se face transmisia. Pentru a îndeplini condiția de timp real complexitatea algoritmului de watermarking ar trebui să fie cât mai redusă. Mai mult, dacă watermark-ul poate fi inserat direct în fluxul comprimat, acest lucru va elimina necesitatea decompresiei și recompresiei complete și, astfel, va reduce efortul de calcul. Această idee a dus la dezvoltarea de scheme de watermarking foarte simple. Exploatarea părților specifice ale standardului de compresie poate conduce la algoritmi foarte eficienți. Un flux video codat MPEG conține o succesiune de cod cu lungime variabilă (VLC). Un watermark poate fi inserat consecvent în flux prin modificarea cuvintelor de cod VLC [155]. Standardul MPEG folosește într-adevăr cuvinte de cod VLC asemănătoare, adică cu aceeași lungime run length, aceeași lungime VLC și o diferență de nivel de cuantizare de unu. Astfel de cuvinte de cod VLC pot fi folosite alternativ pentru a ascunde un bit.

5.2.1. Doar un Alt Sistem de watermarking (JAWS)

Când este vorba despre procesare în timp real, algoritmul dezvoltat de Philips Research este deseori considerat de referință. Algoritmul JAWS a fost inițial dezvoltat pentru monitorizarea emisiei și este unul dintre candidații de frunte pentru watermarking în formatul

DVD. Cerința de procesare în timp real este îndeplinită prin folosirea de operații simple la rata cadrelor și doar câteva mai complexe la o rată mai mică decât rata cadrelor [156].

Procesul de inserare este prezentat în Figura 5.1 și funcționează astfel:

- Se generează o secvență de referință p_r de dimensiune $M \times M$ distribuită normal.
- Se creează un watermark de referință w_r conform relației (5.1):

$$w_r = p_r - \text{trans}(p_r, \text{mesaj}) \quad (5.1)$$

unde funcția $\text{trans}(\cdot)$ returnează o versiune translatată ciclic a secvenței de referință p_r .

În sistemul JAWS mesajul este codat complet prin translatarea celor două secvențe de referință. Această secvență este apoi multiplicată, posibil și cu trunchiere, pentru a obține watermark-ul complet w .

- Pentru fiecare cadru, acest watermark este modificat din punct de vedere perceptual astfel încât inserarea să rămână imperceptibilă. Fiecare element i al watermark-ului este scalat cu factorul de activitate locală $\lambda(i)$ a cadrului obținut prin filtrare laplaciiană. Cu cât regiunea este mai uniformă, cu atât este mai mic factorul de activitate locală. Acest lucru este în concordanță cu faptul că ochiul uman este mai sensibil la zgomotul ce apare în regiuni uniforme ale cadrului.
- Watermark-ul este, în cele din urmă, scalat cu un factor global s și însumat cadrului F pentru a obține cadrul cu watermark F_w . Procesul total de inserare poate fi exprimat astfel:

$$F_w(i) = F(i) + s\lambda(i)w(i) \quad (5.2)$$

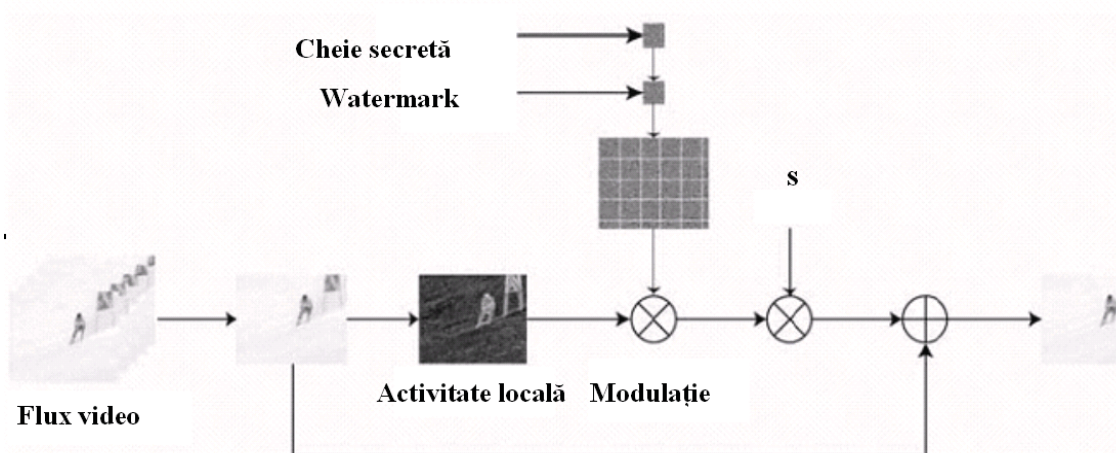


Figura 5.1. Descrierea inserării JAWS

La detector, cadrele sunt sumate și stocate într-un buffer B de dimensiune $M \times M$. Detectorul caută apoi toate aparițiile secvenței de referință p_r în buffer printr-o convoluție ciclică bi-dimensională. Deoarece o astfel de operație se realizează cel mai eficient în domeniul frecvență, se obține o detecție prin filtrare SPOMF (Symmetrical Phase Only Matched Filtering), dată de relația (5.3):

$$SPOMF(B, p_r) = IFFT[\varphi(FFT(B)) \cdot \varphi(FFT(p_r))^*] \quad (5.3)$$

$$\varphi(x) = \begin{cases} x / |x|, & \text{daca } x \neq 0 \\ 1, & \text{daca } x = 0 \end{cases} \quad (5.4)$$

unde $FFT(\cdot)$ și $IFFT(\cdot)$ sunt Transformata Fourier Rapidă directă, respectiv inversă și x^* reprezintă conjugarea complexă. Figura 5.2 arată rezultatele unei astfel de detecții. Pot fi izolate două maxime ce corespund celor două apariții ale lui p_r în w_r . Datorită posibilității apariției jitter-ului pozițional, nu pot fi folosite toate pozițiile relative dintre vârfuri, ele sunt impuse a fi multiplii ai pasului G al grilei. După ce detectorul a extras vârfurile, informația ascunsă poate fi extrasă ușor. Se observă că această schemă este invariantă la translație, deoarece o translație nu modifică poziția relativă dintre vârfuri.

Ulterior, această schemă a fost îmbunătățită. De exemplu, invarianța la translație a fost exploatată mai departe pentru a crește cantitatea de informație ascunsă [157], iar simple modificări au permis obținerea invarianței la scalare [158].

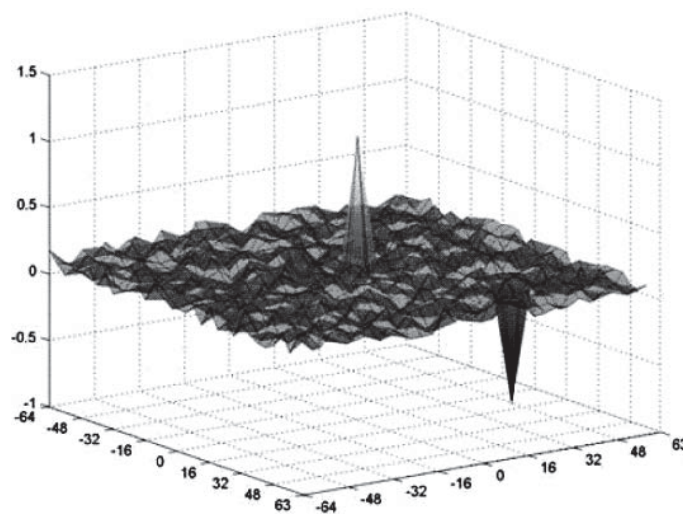


Figura 5.2. Exemplu de detecție SPOMF

5.3. Direcții principale în watermarking-ul video

Watermarking-ul video este un domeniu relativ nou de cercetare ce beneficiază de rezultatele pentru imagini. Au fost propuși mulți algoritmi în literatura științifică și pot fi izolate trei mari tendințe. Cea mai simplă și directă abordare este considerarea unei secvențe video ca o succesiune de imagini. Alt punct de vedere consideră și exploatează dimensiunea temporală adițională pentru a dezvolta algoritmi noi și mai robuști de watermarking video. Ultima direcție de cercetare consideră secvența video ca niște date comprimate corespunzător unui standard de compresie video și caracteristicile unui astfel de standard pot fi folosite pentru a obține o schemă eficientă de watermarking. Fiecare dintre aceste abordări are puncte forte și puncte slabe (vezi Tabelul 5.1).

Tabelul 5.1. Avantajele și dezavantajele tendințelor în watermarking-ul video

Clasificare	Avantaje	Dezavantaje
Adaptare imagine - video	Beneficiază de toate rezultatele pentru imagini	Pot necesita un efort mare de calcul
Integrarea dimensiunii temporale	Algoritmi video care de obicei permit o mai mare robustețe	Pot necesita un efort mare de calcul
Exploatarea standardului de compresie	Algoritmi simplii, ce permit procesarea în timp real	Watermark-ul este legat doar de respectivul standard de compresie

5.3.1. Adaptarea tehnicilor folosite la imagini pentru video

La început, domeniul watermarking-ului digital a fost investigat pentru imagini. Au fost descoperiți o serie de algoritmi și rezultate și, atunci când a început investigarea de noi domenii, cum ar fi cel video, prima idee a fost încercarea refolosirii rezultatelor descoperite anterior. În consecință, comunitatea watermarking a considerat la început un semnal video ca o succesiune de imagini și a adaptat scheme de watermarking deja existente pentru imagini la video. Exact același fenomen a avut loc când comunitatea ce acționa în domeniul codării a trecut de la codarea de imagini la codarea video. Primul algoritm propus pentru codarea video a fost JPEG în Mișcare (Moving JPEG – M-JPEG), care, pur și simplu, comprima fiecare

cadru al unei secvențe video cu standardul JPEG de compresie a imaginilor. Cel mai simplu mod de a extinde o schemă de watermarking pentru imagini este inserarea aceluiași watermark în cadrele video cu o anumită frecvență. Dacă secvența video conține watermark-ul, atunci la ieșirea detectorului ar trebui să se observe un răspuns periodic cu frecvența de inserare [159]. O astfel de schemă nu are însă prea multe avantaje. Detectorul poate doar verifica dacă un watermark dat este prezent sau nu, dar nu poate extrage mesajul ascuns. Pe de altă parte, datele gazdă sunt mult mai mari decât în cazul imaginilor. Deoarece ar trebui să fie posibil să ascundem mai mulți biți într-un semnal gazdă de dimensiuni mai mari, ar putea fi așteptate watermark-uri cu adaos mare de informație. Acest lucru poate fi realizat ușor prin inserarea unui watermark independent, cu mulți biți, în fiecare cadru al secvenței [160]. Ceea ce trebuie însă avut în vedere este că adaosul de informație este contrabalansat de pierderea în robustețe.

Watermark cu energie diferită (WED)

Metoda WED a fost inițial dezvoltată pentru imagini și a fost extinsă la video prin adăugarea de watermark-uri în cadrele I ale fluxului MPEG [155]. Metoda este bazată pe eliminarea selectivă a coeficienților DCT de frecvență înaltă din fluxul de date comprimate. Procesul de inserare este prezentat în Figura 5.3. Blocurile de 8x8 pixeli ale cadrului video sunt mai întâi amestecate pseudo-aleator.

Această operație creează cheia secretă a algoritmului și aleatorizează în domeniul spațial aranjarea blocurilor, adică distruge corelația spațială între blocuri învecinate. Cadrul obținut este apoi împărțit în n blocuri de dimensiune 8 x 8 pixeli. În Figura 5.3 n este egal cu 16. Un bit de watermark este inserat în fiecare dintre aceste blocuri prin introducerea unei diferențe de energie între coeficienții DCT de frecvențe înalte din jumătatea de sus a blocului (regiunea A) și cei din jumătatea de jos a blocului (regiunea B). Din acest motiv metoda se numește watermark cu energie diferită.

Pentru a introduce diferența de energie, pentru fiecare din cele n blocuri de 8 x 8 pixeli este calculat blocul DCT corespunzător și coeficienții DCT sunt precuantizați cu factorul de calitate Q_{JPEG} folosind procedura de cuantizare JPEG standard. Coeficienții obținuți sunt împărțiți în două jumătăți și energia de înaltă frecvență este calculată pentru fiecare regiune conform următoarei relației (5.5):

$$E(c, n, Q_{JPEG}) = \sum_{b=0}^{n/2-1} \sum_{i \in S(c)} ([\theta_{i,b}]_{Q_{JPEG}})^2 \quad \text{cu } S(c) = \{i \in \{0, 63\} \mid (i > c)\} \quad (5.5)$$

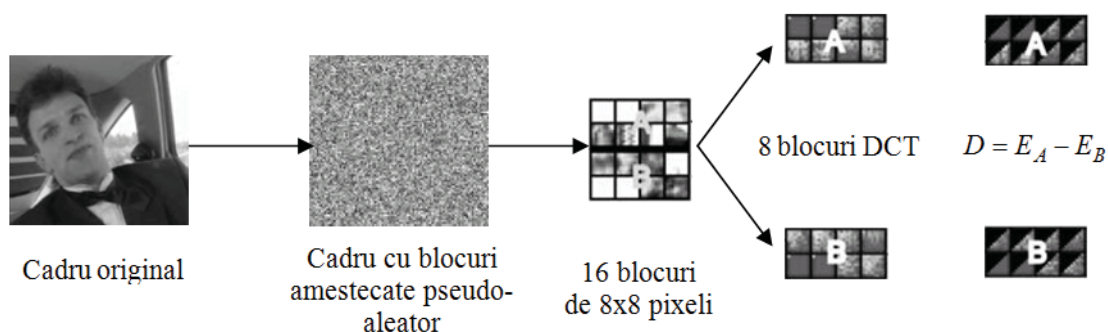


Figura 5.3. Inserarea WED

unde $\theta_{i,b}$ este coeficientul DCT cu indicele i în ordinea zig-zag din blocul DCT cu numărul b . $[\cdot]$ reprezintă precuantizarea cu factorul de calitate Q_{JPEG} și c este un indice de tăiere dat care în Figura 5.3 a fost setat la valoarea 27. Valoarea bitului de inserat este codată în semnul diferenței de energie $D = E_A - E_B$ dintre cele două regiuni A și B. Toată energia după indicele de tăiere c este eliminată în ambele regiuni prin setarea coeficienților corespunzători la zero pentru a obține semnul corespunzător pentru diferența D . Ar trebui observat faptul că acest lucru se poate face ușor direct în domeniul comprimat prin translatarea marker-ului End Of Block (EOB) al blocurilor corespunzătoare spre coeficientul DC până la indicele de tăiere. În cele din urmă, se calculează DCT inversă și aleatorizarea blocurilor este inversată pentru a obține cadrul cu watermark.

La detector, diferența de energie este calculată și bitul inserat este determinat în funcție de semnul diferenței D . Acest algoritm a fost îmbunătățit în continuare pentru a adapta indicele de tăiere c la conținutul în frecvență a celor n blocuri 8×8 considerate și astfel încât diferența de energie D să fie mai mare decât un prag D_{dorit} dat de [161].

5.3.2. Integrarea dimensiunii temporale

Principalul dezavantaj în considerarea unei secvențe video ca fiind o succesiune de imagini independente este faptul că nu se ia în seamă noua dimensiune, și anume cea temporală. Codarea a făcut un mare pas înainte prin încorporarea dimensiunii temporale în schemele de codare și, de aceea, este evident că și domeniul watermarking-ului poate profita în egală măsură de avantajele acestei abordări. Mulți cercetători au investigat posibilitatea reducerii impactului vizual al imaginilor datorat watermark-ului prin utilizarea proprietăților sistemului vizual uman, ca de exemplu mascarea în frecvență, mascarea de luminanță și

mascarea de contrast. Astfel de studii pot fi ușor exportate spre domeniul video printr-o abordare cadru cu cadru. Totuși, watermark-ul obținut nu este optim din punct de vedere al vizibilității, deoarece nu are în vedere sensibilitatea temporală a ochiului uman.

Mișcarea este o caracteristică specifică pentru video, deci trebuie dezvoltate măsurări perceptuale specifice secvențelor video pentru a fi exploatate în domeniul watermarking-ului digital. Acest exemplu simplu arată că dimensiunea temporală este un factor hotărâtor în domeniul video, ce trebuie avut în vedere în dezvoltarea de algoritmi eficienți.

a) *Semnalul video privit ca un semnal unidimensional*

Una dintre lucrările de referință în domeniul watermarking-ului digital consideră semnalul video ca un semnal uni-dimensional [162]. Altfel spus, algoritmul nu ia în calcul dimensionalitatea spațială și temporală a semnalului video și îl privește ca o colecție de eșantioane. Un astfel de semnal este achiziționat printr-o simplă scanare pe linii (vezi Figura 5.4). Algoritmul este detaliat în cele ce urmează.

Watermarking video cu spectrul împrăștiat

Fie $a(j) \in \{-1,1\}$ secvența de biți ai watermark-ului, ce trebuie inserat. Această secvență este împrăștiată cu un factor cr conform relației (5.6):

$$b(i) = a(j), \quad j \cdot cr \leq i \leq (j+1) \cdot cr, \quad i \in N \quad (5.6)$$

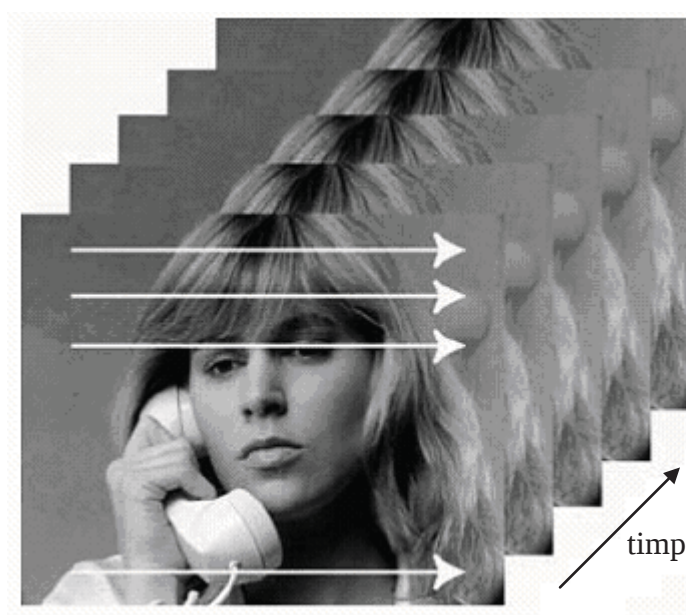


Figura 5.4. Scanarea pe linii într-o secvență video

Operația de împrăștiere permite adăugarea de redundanță prin inserarea unui bit de informație în cr eşantioane de semnal video. Secvența $b(i)$ obținută este apoi amplificată local cu un factor reglabil $\lambda(i) \geq 0$ și modulată cu o secvență binară pseudo-aleatoare $p(i) \in \{-1, 1\}$. În cele din urmă, watermark-ul cu spectru împrăștiat $w(i)$ este adăugat semnalului video $v(i)$ scanat pe linii. Astfel, se obține semnalul video cu watermark $v_w(i)$. Procesul complet de inserare este descris riguros de relația (5.7):

$$v_w(i) = v(i) + w(i) = v(i) + \lambda(i)b(i)p(i), \quad i \in N \quad (5.7)$$

Factorul reglabil $\lambda(i)$ poate fi ajustat în funcție de proprietățile locale ale semnalului video, ca de exemplu mascarea spațială și temporală a SVU, sau poate fi păstrat constant în funcție de aplicație.

Pe partea de detecție, extragerea watermark-ului se realizează ușor printr-o simplă corelație. Totuși, pentru a reduce interferența dintre watermark și semnalele video, secvența video cu watermark este filtrată trece-sus, obținându-se astfel semnalul $v_w(i)$ în care componente majore ale semnalului video inițial sunt izolate și eliminate. Al doilea pas este demodularea. Semnalul video filtrat este multiplicat cu zgomotul pseudo-aleator $p(i)$ folosit la inserare și se calculează corelația $s(j)$ pentru fiecare bit j al watermark-ului:

$$s(j) = \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} p(i) \underline{v_w(i)} = \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} p(i) \underline{v(i)} + \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} p(i) \underline{\lambda(i)b(i)p(i)} \quad (5.8)$$

Corelația conține doi termeni Σ_1 și Σ_2 . Scopul principal al filtrării a fost păstrarea lui Σ_2 neafectat și reducerea lui Σ_1 la 0. În concluzie, corelația devine:

$$s(j) \simeq \Sigma_2 \simeq \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} p(i)^2 \lambda(i)b(i) = a(j) \cdot cr \cdot med(\lambda(i)) \quad (5.9)$$

Bitul ascuns este dat direct de semnul lui $s(j)$. Această tehnică oferă un cadru foarte flexibil pentru o schemă de watermarking mai elaborată.

b) Semnalul video privit ca un semnal temporal

Chiar dacă metoda de watermarking cu spectru împrăștiat oferă un algoritm flexibil ce poate fi folosit pentru scheme de watermarking mai elaborate, ea ignoră complet dimensiunea temporală. Din această cauză watermark-ul inserat nu va fi optim din punctul de vedere al

invizibilității și rezistenței la atacuri. Au fost investigate o serie de alte metode pentru a integra și dimensiunea temporală, iar una dintre ele este interpretarea semnalului video ca un semnal pur temporal. Un exemplu tipic ar fi inserarea unui semnal de watermarking temporal în anumiți pixeli specificați de o cheie [163]. Pentru a asigura invizibilitatea watermark-ului locațiile de inserare ar trebui alese cu grijă. Într-adevăr, chiar dacă modificarea unui singur pixel dintr-o zonă cu textură este imperceptibilă în fiecare cadru în parte, ea ar putea fi vizibilă la redarea secvenței video. În practică se observă că pixelii ce se modifică rapid în timp sau pixelii de la extremitățile regiunilor fără mișcare sunt candidați buni pentru inserare. Oricum, folosirea unui număr mic de pixeli pentru inserarea watermark-ului reduce drastic capacitatea de inserare. De fapt, de multe ori se preferă calcularea unei transformate pe întreaga secvență video pentru a avea un spațiu de inserare mai larg. În particular, descompunerea wavelet temporală poate fi utilă pentru a obține o reprezentare temporală multi-rezoluție compactă a secvenței video [164]. Folosind o astfel de descompunere se poate izola o componentă statică (fără mișcare) și o serie de componente dinamice (cu mișcare). Caracterul multi-rezoluție al Transformatei Wavelet permite watermark-ului existența pe mai multe scări temporale. De exemplu, dacă watermark-ul este inserat în cadrul wavelet de cea mai joasă frecvență temporală (DC), va exista în toate cadrele secvenței video. O altă transformată temporală promițătoare este Analiza Componentelor Independente (ACI). Această transformată produce un set de cadre care pot fi folosite ca surse independente pentru generarea secvenței video procesate [165].

c) Semnalul video privit ca un semnal tri-dimensional

Semnalul video mai poate fi privit și ca semnal tri-dimensional. Acest punct de vedere a fost deja investigat în codare și poate fi extins și asupra watermarking-ului. Transformata Fourier Discretă 3-D poate fi folosită ca o reprezentare alternativă a semnalului video [166]. În acest caz sunt avute în vedere frecvențele spațiale sau temporale medii pentru inserarea watermark-ului pentru a obține un compromis între invizibilitatea watermark-ului și robustețe. Pe de o parte, sistemul vizual uman este avut în vedere pentru a defini o regiune de inserare care să nu conducă la un watermark vizibil, iar, pe de altă parte, regiunea de inserare obținută este modificată, astfel încât să devină imună la compresia MPEG. Pentru a produce un watermark robust au fost investigate și Transformata Wavelet 3-D și Transformata Gabor 3-D [167]. Totuși, considerarea semnalului video ca un semnal tridimensional poate fi imprecisă. Cele trei dimensiuni avute în vedere nu sunt, de fapt, omogene: există două dimensiuni

spațiale și una temporală. Acest aspect și costul de calcul au îngreunat cercetările ulterioare în această direcție. Oricum, această abordare rămâne pertinentă în unele cazuri. În domeniul medical de exemplu, diferitele imagini de pe scanner pot fi cadre diferite ale unei secvențe video. În acest caz, cele trei dimensiuni sunt omogene și poate fi folosită o transformată 3-D.

5.3.3. Exploatarea formatelor de compresie video

Direcții noi de cercetare consideră datele video ca niște date comprimate cu ajutorul unui standard specific de compresie video. Într-adevăr, în cele mai multe cazuri o secvență video este stocată într-o versiune comprimată pentru a ocupa mai puțin spațiu de stocare. Astfel au fost dezvoltate metode de watermarking care inserează watermark-ul direct în fluxul video comprimat.

Watermarking-ul în fluxul comprimat poate fi privit ca o formă de editare video în domeniul comprimat [168]. O astfel de editare nu este ceva obișnuit în practică și apar o serie de probleme.

a) *Modificarea coeficienților în domeniul transformatei*

În [169] este propusă o metodă ce adaugă o secvență pseudo-aleatoare, transformată DCT direct în coeficienții DC-DCT ai unui flux video codat MPEG. Procesul de watermarking ia în considerare doar valorile de luminanță ale cadrelor I. Pentru a insera watermark-ul este folosită următoarea procedură:

- Este generată o secvență pseudo-aleatoare conținând întregii $\{-1,1\}$ folosind o cheie secretă. Această secvență are aceeași dimensiune ca și cadrul I.
- Secvența este modulată de un șir de biți de watermark și multiplicată cu un factor de câștig.
- Este aplicată Transformata DCT 8x8 pe secvența modulată și coeficienții DC rezultați sunt însumați la valorile DC corespunzătoare din toate cadrele I
- Watermark-ul poate fi detectat folosind tehnici de corelație în domeniul DCT sau în domeniul spațial.

Autorii raportează că algoritmul scade drastic calitatea vizuală a secvenței video. De aceea, factorul de câștig al watermark-ului trebuie ales foarte mic (<1) și numărul de pixeli per bit de watermark trebuie ales foarte mare ($>>100.000$) pentru a menține o calitate video rezonabilă pentru secvența video rezultantă. Acest lucru se datorează, în primul rând, faptului

că secvența watermark este inserată în doar unul din cei 64 de coeficienți DCT, coeficientul DC. Mai mult, secvența conține doar componente de frecvență joasă la care ochiul uman este destul de sensibil.

În [162] este propusă o metodă mai sofisticată ce inserează watermark-ul nu doar în coeficienții DC, ci și în coeficienții AC ai fiecărui cadru I, P sau B. Și aici, watermark-ul este o secvență pseudo-aleatoare de întregi $\{-1,1\}$ generați cu ajutorul unei chei secrete. Secvența are aceeași dimensiune ca și cadrul video. Secvența este modulată de un șir de biți de watermark și multiplicată cu un factor de câștig k .

Pentru inserarea watermark-ului, secvența $W(x,y)$ este divizată în blocuri de 8×8 . Aceste blocuri sunt transformate în domeniul DCT și notate $W_{x,y}(u,v)$, unde $x,y = 0,8,16,\dots$ și $u,v = 0 \dots 7$. Apoi blocurile 2-D $W_{x,y}(u,v)$ sunt reordonate prin scanare în zig-zag și devin șiruri $W_{x,y}(i)$, unde $i = 0 \dots 63$. $W_{x,y}(0)$ reprezintă coeficientul DC, iar $W_{x,y}(63)$ este coeficientul AC de frecvență cea mai mare a blocului de watermark. Deoarece blocurile MPEG de dimensiune 8×8 corespunzătoare sunt codate în același mod ca și $W_{x,y}(i)$, aceste șiruri $I_{x,y}(i)$ pot fi direct folosite pentru adăugarea watermark-ului. Pentru fiecare bloc video $I_{x,y}(i)$ din cadrele I, P sau B sunt aplicați următorii pași:

1. Coeficientul DC este modulată astfel:

$$I_{W_{x,y}}(0) = I_{x,y}(0) + W_{x,y}(0) \quad (5.10)$$

ceea ce înseamnă că valoarea medie a blocului de watermark este adăugată la valoarea medie a blocului video.

2. Pentru modularea coeficienților AC în șirul de biți a blocului video codat este căutat următorul cuvânt de cod VLC, reprezentând următorul coeficient DCT diferit de zero. Lungimea și nivelul acestui cuvânt de cod sunt decodate pentru a determina poziția i din scanarea zig-zag și amplitudinea $I_{x,y}(i)$.

3. Apoi este inserat un watermark într-un coeficient AC astfel:

$$I_{W_{x,y}}(i) = I_{x,y}(i) + W_{x,y}(i), \quad i \neq 0 \quad (5.11)$$

Acum intră în calcul și constrângerea, ca rata de bit a secvenței video să nu crească.

Lungimea S_{z_I} a codului VLC necesar pentru a coda $I_{x,y}(i)$ și lungimea $S_{z_{I_w}}$ pentru a coda

$I_{w_{x,y}}(i)$ sunt determinate folosind tabelele VLC B.14 și B.15 ale standardului MPEG-2 [171]. Dacă lungimea codului VLC pentru codarea coeficientului cu watermark este egală sau mai mică decât lungimea codului VLC existent, codul VLC existent este înlocuit. Altfel, codul VLC este lăsat nemodificat. Astfel, coeficientul $I_{x,y}(i)$ este modulat în următorul mod:

$$\begin{aligned} \text{dacă } S_{z_{I_w}} \leq S_{z_I} \quad \text{atunci} \quad I_{w_{x,y}}(i) &= I_{x,y}(i) + W_{x,y}(i) \\ \text{altfel} \quad I_{w_{x,y}}(i) &= I_{x,y}(i) \end{aligned} \quad (5.12)$$

4. Această procedură este repetată până când sunt procesați toți coeficienții AC ai blocului video codat.

Pentru extragerea informației watermark, fluxul video MPEG este întâi decodat complet și biții de watermark sunt extrași prin corelația dintre cadrele decodate și secvența de watermark $W(x, y)$ în domeniul spațial folosind tehnicile standard descrise în capitolul pentru imagini.

Un alt exemplu este algoritmul anterior de watermark cu spectru împrăștiat, ce a fost adaptat astfel încât watermark-ul să poate fi inserat direct în coeficienții DCT diferiți de zero ai unui flux MPEG [162]. Prima grijă a fost ca procesul de inserare a watermark-ului să nu crească rata de bit de ieșire. În primul rând, practic nimic nu ne asigură că un coeficient DCT cu watermark va fi codat cu lungime variabilă (VLC) cu același număr de biți ca și cel fără watermark. O strategie directă ar fi, atunci, inserarea watermark-ului doar în coeficienții care nu necesită mai mulți biți pentru codarea VLC. Un al doilea aspect a fost prevenirea propagării distorsiunilor datorate watermark-ului de la un cadru la altul. Standardul MPEG are la bază predicția mișcării și orice distorsiune se propagă în cadrele învecinate. Pentru a preveni ca acumularea de astfel de semnale să producă o calitate video slabă, poate fi adăugat un semnal compensator. Într-un astfel de caz compensarea mișcării poate fi văzută ca o constrângere. Dar, pe de altă parte, poate fi exploatată, astfel încât vectorii de mișcare ai fluxului MPEG să poarte informația ascunsă [172]. Componentele vectorului de mișcare pot fi cuantizate conform unei reguli, ce depinde de bitul ce trebuie inserat. De exemplu, componenta orizontală a vectorului de mișcare poate fi cuantizată la o valoare pară de pași de cuantizare, dacă bitul de inserat este 0, și la o valoare impară, dacă bitul este 1.

Nu toate cadrele unei secvențe video codate MPEG sunt codate în același fel. Cadre codate intra (I) sunt comprimate cu standardul JPEG de compresie a imaginilor iar cadrele

codate inter (B și P) sunt codate prin predicție folosind alte cadre video învecinate. Ca o consecință, pot fi folosite strategii alternative de watermarking în funcție de tipul cadrelor ce urmează să conțină informația [173].

b) Modificarea vectorilor de mișcare

Un alt element cheie în codarea video este estimarea/compensarea mișcării pentru a reduce redundanța temporală. Într-adevăr, cadrele video succesive sunt foarte asemănătoare iar codarea video are ca scop de bază predicția unui cadru dintr-un alt cadru folosind predicția mișcării pentru a reduce cantitatea de date transmise. De exemplu, în standardul MPEG există o distincție clară între cadre I ce sunt codate ca niște imagini de sine stătătoare și cadre P sau B care sunt codate ca referință la un cadru I sau alt cadru P, respectiv ca referință la două cadre, fiecare I sau P. Se obțin o serie de vectori de mișcare ce sunt transmiși la decodor pentru a realiza compensarea mișcării. Ar fi interesant de folosit acești vectori de mișcare ca și potențiali candidați pentru inserarea unui watermark. În acest sens, se poate impune o regulă de paritate pentru componentele vectorului de mișcare [172]. The exemplu, componenta orizontală a unui vector de mișcare este cuantizat la o valoare pară, dacă bitul watermark-ului de inserat este 0 și la o valoare impară altfel. Din motive de vizibilitate se pot considera pentru inserare doar vectorii de mișcare de valori mari și se poate modifica fie componenta orizontală, fie cea verticală a vectorului de mișcare în funcție de unghiul vectorului [174]. Ca alternativă, schemele de cuantizare pot fi, de asemenea, folosite pentru modificarea informației de mișcare. În acest sens, vectorii de mișcare pot fi cuantizați folosind o grilă pătrată, circulară sau unghiulară [175, 176]. Astfel de abordări s-au dovedit a fi puțin mai robuste. Oricum, una dintre principalele preocupări la modificarea informației de mișcare este fidelitatea: este foarte dificil de calculat care este impactul perceptual al modificării vectorilor de mișcare. Totuși, această problemă pare a nu fi critică pentru unele aplicații. De exemplu, informația de mișcare poate fi modificată pentru a realiza o criptare parțială, numită și scrambling de watermark [177]. În acest sens, scopul este degradarea calității video, dar totuși permițând utilizatorului privat perceperea conținutului video, dându-i o idee despre conținutul original pentru a produce un impuls de cumpărare.

c) Modificarea cuvintelor de cod VLC

Multe codoare realizează de obicei cuantizarea coeficienților din domeniul transformat și a vectorilor de mișcare folosind fie o cuantizare scalară, fie una vectorială. Apoi informația

rezultată este reprezentată prin diferite simboluri care sunt trimise unui codor entropic pentru a obține șirul de biți final. De exemplu, la standardul MPEG coeficienții DCT sunt scanați în zigzag și reprezentați sub forma de tuple (run, level - lungime, nivel). Lungimea este egală cu numărul de zerouri ce preced un coeficient nenul, iar nivelul este egal cu valoarea coeficientului cuantizat. Aceste tuple sunt trimise unui codor entropic. În practică, în standardul MPEG sunt definite tabele de căutare pentru asocierea fiecărei tuple posibile unui cuvânt de cod codat cu lungime variabilă (VLC). Astfel cercetători au investigat cum se poate modifica direct fluxul de biți, respectiv cuvintele de cod VLC pentru a evita compresia și decompresia completă ce sunt consumatoare de timp. În acest sens, o lucrare deschizătoare de drumuri a identificat un set de cuvinte VLC ce poate fi modificat fără introducerea de artefacte vizuale prea mari [155]. Acest algoritm este detaliat mai jos. Chiar dacă au fost propuse câteva variații ale acestei abordări [178], cea mai importantă direcție de cercetare este cea care face cuvintele de cod VLC să prezinte proprietăți de resincronizare după ce sunt afectate de erori de bit [179, 180]. Astfel de cuvinte de cod VLC sunt numite cuvinte VLC reversibile (RVLC) și pot fi decodate în două moduri. Ideea este să se folosească apoi puterea de corecție de erori a acestor RVLC pentru conceperea de scheme reversibile de watermarking: modificările binare datorate procesului de watermarking sunt considerate erori de canal și pot fi corectate.

Tabelul 5.2. Exemple de cuvinte VLC 1c din Tabelul B.14 al standardului MPEG.

Cod cu lungime variabilă	Lungime VLC	Run	Level	LSB al Level
0010 0110 s	8+1	0	5	1
0010 0001 s	8+1	0	6	0
0000 0001 1101 s	12+1	0	8	0
0000 0001 1000 s	12+1	0	9	1
0000 0000 1101 0 s	13+1	0	12	0
0000 0000 1100 1 s	13+1	0	13	1
0000 0000 0111 11 s	14+1	0	16	0
0000 0000 0111 10 s	14+1	0	17	1
0000 0000 0011 101 s	15+1	1	10	0
0000 0000 0011 100 s	15+1	1	11	1
0000 0000 0001 0011 s	16+1	1	15	1
0000 0000 0001 0010 s	16+1	1	16	1

Modificarea bitului de paritate în domeniul VLC

În unele lucrări din literatura de specialitate s-a modificat direct fluxul video comprimat pentru a insera un watermark. Astfel de algoritmi sunt foarte interesanți datorită ratei mari de inserare și a complexității de calcul scăzute obținute. În cazul unui flux video MPEG un watermark conținând l biți b_i ($i=0,1,\dots,l-1$) este inserat în fluxul video prin selectarea cuvintelor VLC potrivite și egalarea bitului celui mai puțin semnificativ (LSB) al pasului de cuantizare corespunzător cu biții watermark-ului [155]. Pentru asigurarea invizibilității perceptuale și a dimensiunii fluxului video doar câțiva coeficienți VLC numiți cuvinte VLC etichetă (lc-VLC – label bit carrying VLC) sunt avuți în vedere pentru inserarea watermark-ului. Aceste cuvinte VLC au proprietatea interesantă că există un alt cuvânt VLC care are:

- aceeași valoare run (lungime)
- o diferență de level (nivel) de 1
- aceeași lungime VLC

Conform tabelelor B.14 și B.15 ale standardului MPEG, există multe astfel de cuvinte VLC, câteva exemple fiind date în Tabelul 5.2. Simbolurile reprezintă bitul de semn care indică semnul nivelului coeficientului DCT. Pentru inserarea biților de watermark într-un flux video MPEG, sunt testate toate cuvintele VLC din toate macro-blocurile, exceptând coeficienții DC din motive de vizibilitate. Dacă este găsit un cuvânt lc-VLC și LSB-ul nivelului său nu este egal cu bitul b_i al watermark-ului, atunci acest cuvânt VLC este înlocuit cu altul, al cărui nivel are LSB-ul egal cu bitul de watermark de inserat. Altfel, dacă LSB-ul cuvântului lc-VLC original este egal cu bitul b_i al watermark-ului, cuvântul VLC nu este modificat. Această procedură este repetată până la inserarea tuturor biților watermark-ului. În Figura 5.5 este prezentat acest proces pentru 3 biți de watermark inserați într-un flux MPEG. La detector biții watermark-ului sunt extrași testând toate cuvintele VLC din fiecare macro-bloc. Dacă este găsit un lc-VLC, este extras LSB-ul său și adăugat șirului de biți ai watermark-ului. Procedura este repetată până când nu mai sunt găsite cuvinte lc-VLC. Chiar dacă astfel de algoritmi sunt destul de sensibili la atacuri prin editare video, ei sunt complet adaptați aplicațiilor de ascundere de date.

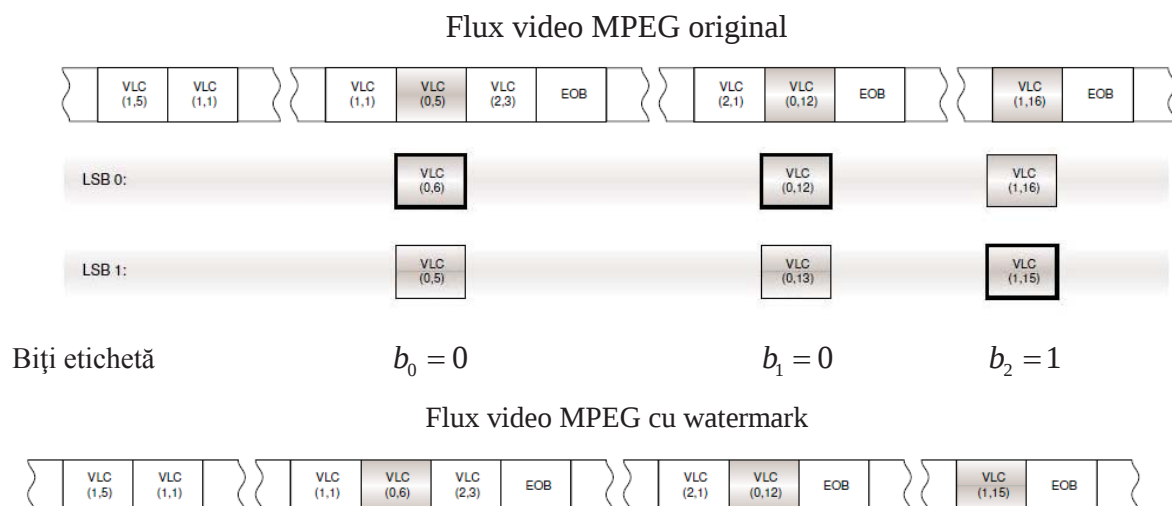


Figura 5.5. Proces de inserare a watermark-ului în cuvinte VLC.

d) Modificarea parametrilor de model ai obiectelor video

Formatul MPEG-4 poate fi, de asemenea, folosit pentru aplicații specifice. MPEG-4 conține, pe lângă compresia video obișnuită, funcționalități adiționale, ca de exemplu animația bazată pe model a suprafeței 3-D a capului uman folosind așa numiții parametrii de animație facială (PAF). Acești parametrii sunt, de exemplu: „rotire cap”, „deschidere gură” sau „ridicarea colțului din dreapta a buzei”. Modelul capului folosit la decodor este fie un cap obișnuit predefinit și un model al feței, fie un model specific ce poate fi transmis folosind așa numiții parametrii de definire facială (PDF).

Unealta de animație a feței permite compresia scenelor cu cap și umeri, ca de exemplu în aplicații de video-telefonie, la o rată de bit de 1000 biți/s. În [181] Hartung și al. propun o metodă de watermarking cu spectru împrăștiat, cu inserarea watermark-ului în parametrii PAF. Watermark-urile sunt inserate aditiv în parametrii de animație. Filtrarea watermark-ului cu spectru împrăștiat prin filtre trece-jos și atenuarea adaptivă a amplitudinii previn distorsiuni ale modelelor animate ale capetelor. Watermark-urile pot fi extrase prin calcularea corelației cu parametrii cu watermark, dar și cu secvențe video ce conțin modele 3-D de capete animate folosind parametrii cu watermark, chiar și după modificări survenite după o compresie bazată pe blocuri. Figura 5.6 arată un exemplu de cadre video dintr-o secvență randată folosind un model 3-D și parametrii de animație. În acest caz, parametrii trebuie să fie mai întâi estimați din secvență. O observație interesantă este că watermark-ul nu este conținut în reprezentarea obiectului vizualizat (în pixeli), ci în semantică (modul în care se mișcă capul și fața).



Figura 5.6. Cadru cu watermark dintr-o secvență video randată folosind (a) un model 3-D al capului și parametrii de animație și (b) un cadru similar după compresie video MPEG-2 la 600 kbit/s

Inserarea watermark-ului direct în domeniul fluxului video comprimat permite de multe ori procesarea video în timp real. Totuși, dezavantajul constă în faptul că watermark-ul este legat definitiv de standardul de compresie folosit și nu supraviețuiește unei conversii de format.

5.4. Concluzii

Watermarking-ul digital a fost extins de la imagini la conținut video. Cercetări viitoare în acest domeniu sunt motivate puternic de nevoia crescândă a deținătorilor proprietății intelectuale de a-și proteja drepturile. Datorită mizei economice mari, watermarking-ul digital este un domeniu de viitor. Apariția de aplicații noi este inherentă, la fel ca și combinarea diferitelor abordări existente. De exemplu, un watermark poate fi separat în două părți: una pentru protecția drepturilor de autor și a doua pentru amprenta clientului. Totuși, mai multe provocări trebuie avute în vedere. Robustețea schemei trebuie abordată cu multă atenție. Există o serie de metode de procesare video neagresive ce ar putea altera semnalul watermark. S-ar putea să nu fie posibilă imunizarea la toate aceste atacuri și vor trebui definite constrângeri corespunzătoare aplicației dorite. În ciuda tuturor acestor probleme, mulți algoritmi au fost deja propuși în literatura de specialitate. Pornesc de la simpla adaptare a

algoritmilor watermarking pentru imagini la conținut video și merg până la scheme de watermarking specifice pentru video.

Mai există încă drumuri deschise în watermarking-ul video. Această tehnologie nu este încă atât de matură ca și cea pentru imagini. Aproape toate procesările posibile au fost investigate pentru imagini. În schimb, algoritmii video au rămas relativ simpli. Multe procesări video nu au fost încă încercate și domeniul e departe de a fi epuizat. Mai mult, există modalități de măsură a calității perceptuale a secvențelor video, dar marea provocare constă în a le putea exploata în timp real.