Sistemi di Elaborazione: Sicurezza su Reti **Anno Acc. 1998-99**
PROF. ALFREDO DE SANTIS

DIGITAL WATERMARKING TECHNOLOGY

a cura di Catalini Fabio, Giacomino Tiziana, Marzano Andrea, Serpico Angela, Sirignano Donatella
12 Marzo 1999

Indice

1 Introduzione

La trasmissione dell'informazione oggi avviene in forme totalmente nuove. Internet, per esempio, può essere visto come un grosso *database* distribuito sul globo terrestre che contiene informazioni di vario tipo in formato *testuale*, *grafico*, *audio* e *video*. Ogni utente connesso alla rete ha la possibilità di recuperare queste informazioni in qualunque parte del mondo si trovi.

La rete, i protocolli e le applicazioni che rendono possibile l'esistenza di un simile database sono il frutto di anni di ricerca: precisamente essi racchiudono gli sforzi che sono stati fatti nelle tecnologie dell'informazione e delle comunicazioni in buona parte del nostro secolo. Complessivamente costituiscono una *struttura informativa* pensata per rendere *condivisibile* grosse quantità di dati multimediali.

Obiettivi principali nella ricerca degli anni passati sono stati quelli dell'individuazione di efficienti *rappresentazioni* e *modalità di trasmissione* delle informazioni che fossero in grado di garantire *affidabilità* e *costi contenuti*, requisiti di base per un utilizzo di massa.

Con l'aumentare delle applicazioni e della quantità di informazioni disponibile in rete sono aumentati gli accessi e gli usi illegali delle stesse.
Nell'ultimo ventennio la *crittografia* ha realizzato sistemi e protocolli in grado di garantire *riservatezza*, *autenticità* ed *integrità* delle comunicazioni e delle informazioni digitali. Esiste però un livello del problema della *protezione dei dati* sul quale poco è stato fatto in passato e soltanto nell'ultimo paio d'anni è stata posta un'attenzione significativa da parte della comunità scientifica. Il seguente esempio aiuta a capire.

Il commerciante $\mathcal{A}$ vende agli acquirenti $\mathcal{B}_1, \ldots, \mathcal{B}_m$ un file grafico di notevole valore $\mathcal{I}$ e glielo invia in forma cifrata tramite la rete.

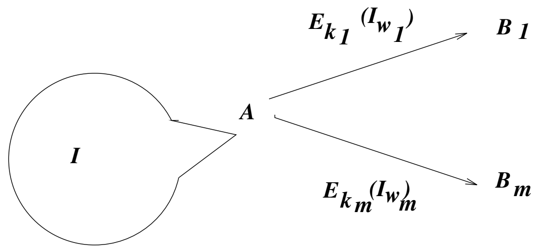

Figura 1: Distribuzione di I agli acquirenti

- Come può $\mathcal{A}$ assicurarsi che $\mathcal{B}_i$, decifrato il file, non lo renda pubblico o lo *rivenda illegalmente* a terzi?
- Esiste un modo per *individuare* le copie illegali?
- Esiste un modo per *rintracciare* gli acquirenti che hanno distribuito illegalmente il file?
- Come può un utente autorizzato *provare* i suoi diritti d'uso sul file acquistato?

e se, ancora, $\mathcal{A}$ fosse un artista che ha realizzato la sua opera in formato digitale

- Come può $\mathcal{A}$ garantire, in modo inequivocabile, la propria *paternità* sul file una volta che è stato reso pubblico evitando che qualcun altro se ne appropri?

- Esiste un modo per *proibire contraffazioni*?

Questi semplici esempi permettono di evidenziare un problema generale: *la mancanza di un'infrastruttura che riesca a garantire la protezione globale dei dati in modo tale che i diritti di "produttori" e "consumatori" di informazioni siano tutelati.*
Occorre cioè realizzare protocolli che riescano a garantire tutte le parti che entrano in gioco nella comunicazione digitale.

La *Watermarking Technology* cerca di fornire soluzioni a questo tipo di problema. L'idea di base è molto semplice: si vuole inserire all'interno del documento una sequenza di bit, w, detta *watermark*, che abbia le seguenti caratteristiche:

- **Invisibile:** la marca deve essere immersa in modo che la sua presenza risulti impercettibile. Ad esempio, se si tratta di immagini, deve accadere che, da un punto di vista visivo l'immagine "con marca" e quella senza marca devono apparire identiche.

- **Legata al documento:** la marca deve essere funzione del documento e parte integrante di esso.

- **Robusta:** la marca deve essere in grado di resistere a tutte le trasformazioni operabili sul documento, purchè quest'ultimo non venga distrutto. Cioè, dato un documento $\mathcal{D}$, indipendentemente dalle trasformazioni che possono essere effettuate su esso, la presenza di w all'interno di $\mathcal{D}$ deve essere sempre reperibile.

In sintesi uno schema di Watermarking stabilisce semplicemente due cose:

- come la stringa w deve essere inserita nel documento;

- come la stringa w deve essere recuperata.

Una introduzione alle problematiche della Digital Watermarking Technology può essere rinvenuta in [1] [2] [3] [4] [5] [6].

2 Ambienti applicativi

Gli *schemi di Watermarking* permettono di perseguire i seguenti obiettivi:

- **Affermazione di paternità :** data un'immagine $\mathcal{I}$, Alice genera un watermark w e lo immerge nella sua immagine originale, poi rende pubblica l'immagine marcata. Nel momento in cui l'utente Bob si appropria di $\mathcal{I}_w$ e afferma di essere il creatore, Alice può dimostrare, con facilità, che l'originale del documento è $\mathcal{I}$ e w è la stringa che ha scelto come marca e quindi $\mathcal{I}_w$ è una *sua* creazione e non di Bob.

- **Individuazione degli impostori e delle falsificazioni:** data un'agenzia $\mathcal{A}$ che distribuisce un file grafico $\mathcal{I}$ a m utenti $\mathcal{B}_1, \ldots, \mathcal{B}_m$, può succedere che, un utente $\mathcal{B}_i$ venda illegalmente il file $\mathcal{I}$. Utilizzando tale tecnologia vediamo cosa si può fare per evitare che ciò accada. L'agenzia $\mathcal{A}$ vende $\mathcal{I}_{wi}$, il file con la marca w_i all'utente $\mathcal{B}_i$ (con i=1,..,m).

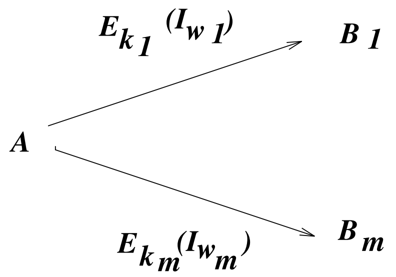

Figura 2: Distribuzione dell'immagine marcata

Nel momento in cui l'utente $\mathcal{B}_j$ rende pubblica la propria immagine $\mathcal{I}_{wj}$, ossia la ridistribuisce illegalmente, l'agenzia $\mathcal{A}$, che ha venduto il documento e ha messo una marca w_i diversa per ogni utente $\mathcal{B}_i$, può andare a vedere nell'immagine pubblica quale marca è presente e quindi accusare in maniera certa l'utente $\mathcal{B}_j$. Ogni utente $\mathcal{B}_i$, con $i \neq j$, ha la possibilità di dimostrare che non è stato lui a rendere pubblica l'immagine, in quanto, la marca presente nel suo documento è diversa rispetto a quella del documento dell'utente $\mathcal{B}_j$.

- **Autenticazione e Verifica dell'integrità del documento:** questi sono problemi già affrontati attraverso tecniche crittografiche, quali le firme digitali e i MAC (Codice di Autentificazione del Messaggio). Con una marca è possibile autenticare il documento e verificarne l'integrità. La differenza rispetto alle tecniche crittografiche sta nel fatto che la marca è presente all'interno dei dati, cioè non è un pezzo di dato che viene attaccato al file, ma un'informazione nascosta all'interno del dato stesso.

- **Aggiungere informazioni varie:** i watermark possono essere usati per inserire informazioni aggiuntive, in un documento, informazioni che possono essere utili in determinati momenti. Per esempio, una macchina fotografica digitale potrebbe "immergere" nelle foto scattate data, ora, e luogo della foto automaticamente.

- **Aggiungere informazioni di controllo:** in particolari sistemi che gestiscono dati digitali, i watermark possono essere immersi nei dati al fine di comunicare le modalità di utilizzazione ai dispositivi hardware che con essi interagiscono. Per esempio di recente è stato commercializzato il Digital Versatile Disk (DVD). Una marca potrebbe essere introdotta nel file per comunicare informazioni sull'eventuale permesso di copia e sull'originalità della copia stessa; un eventuale riproduttore, non appena il disco viene inserito, potrebbe controllare l'originalità della copia e riprodurla, o accorgersi che è una contraffazione e interrompere la riproduzione.

- **Protezione dei contenuti:** i produttori spesso rendono disponibili i loro prodotti per prova, per periodi limitati. I watermark potrebbero essere utilizzati per stabilire il periodo e le restrizioni d'uso.

In seguito ci interesseremo soprattutto di immagini, tuttavia alcune delle idee che verranno esposte sono applicabili anche ad altri tipi di dati.

3 Modello generale

Un'immagine può essere vista come una matrice i cui elementi sono detti pixel (Figura 3). Un pixel è associato ad una stringa binaria di 8, 16, 24 o 32 bit. Il numero di bit utilizzati per descrivere il pixel viene detto *risoluzione in ampiezza* dell'immagine, il numero di righe *risoluzione orizzontale* dell'immagine, il numero delle colonne *risoluzione verticale* dell'immagine.

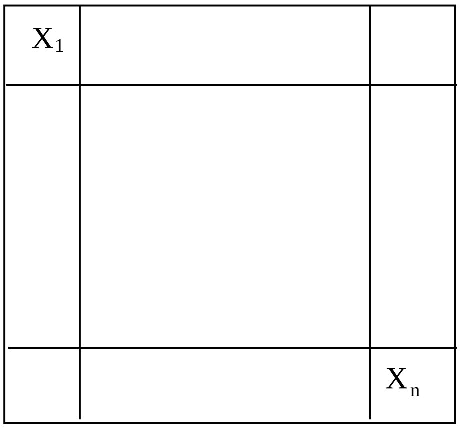

Figura 3: Matrice rappresentante l'immagine

Possiamo immaginare che ogni stringa punti ad una matrice di colori. Supponiamo che la nostra immagine sia a toni di grigio e che ogni pixel sia associato ad una stringa di 8 bit, con i quali si possono rappresentare 256 colori diversi e possiamo pensare di disporli in una matrice detta "matrice dei toni di grigio" (Figura 4).

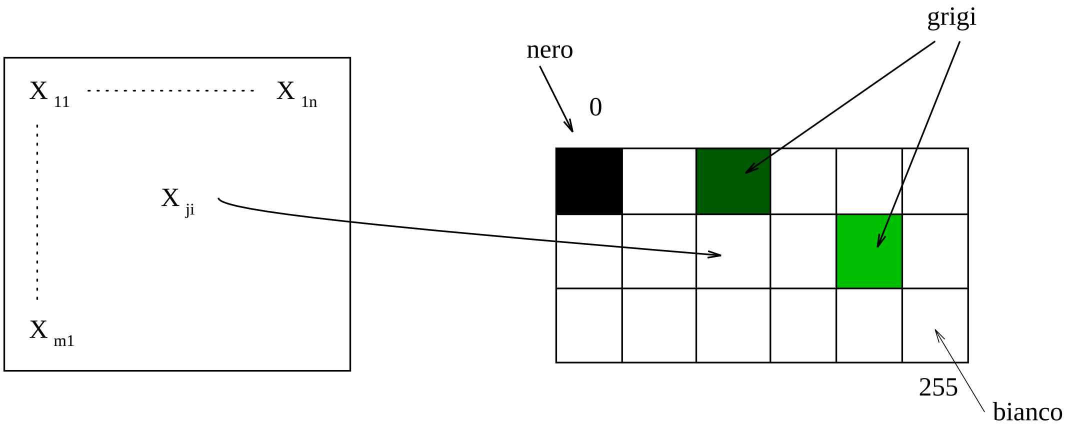

Figura 4: Matrice dei toni di grigio

Le immagini a colori sono anch'esse rappresentate da matrici i cui pixel sono associati a stringhe binarie di 24 bit, suddivise in 3 campi: ognuno di questi campi rappresenta uno dei tre colori fondamentali (rosso, verde e blu) a partire dai quali è possibile derivare tutti gli altri. Nell'associazione del pixel i primi 8 bit vengono considerati come un puntatore ad una

matrice contenente 256 valori di rosso, i secondi 8 bit come un puntatore ad un'altra matrice di 256 valori, che rappresentano i vari livelli di verde ed i rimanenti 8 bit come un puntatore ad una matrice di altri 256 valori che rappresentano i vari livelli di blu.
Un watermark è una funzione di diversi parametri:

- ID, l'identità di colui che inserisce la marca all'interno del documento;
- K, la chiave privata custodita da qualche parte;
- Im, l'immagine all'interno della quale la marca viene inserita;
- AGG, alcune informazioni aggiuntive, ad esempio la taglia dell'immagine.

L'operazione di inserimento immerge il watermark nell'immagine originale, in modo tale che la sua presenza risulti impercettibile. A seconda degli schemi di watermark, la verifica può avvenire per estrazione o per rilevamento. Nel primo caso esiste una procedura che *estrae* esattamente il watermark in essa inserito, nel secondo esiste una procedura che *recupera* la stringa presente all'interno dell'immagine e la confronta con quella originariamente inserita. Se le due stringhe così ottenute non sono molto differenti, tale procedura considera la stringa rinvenuta come il watermark iniziale.
Con schemi del primo tipo, un utente può inserire in un documento un numero seriale ed essere sicuro che, se il documento non viene danneggiato, un qualunque altro utente dotato della procedura di estrazione potrà leggerla. I secondi, invece, in fase di rilevamento, assicurano soltanto che nel documento sia presente il watermark w inserito in precedenza. La probabilità dell'evento dipende dal grado di fedeltà della procedura di rilevamento.
Precisamente, supponiamo che l'utente $\mathcal{A}$ abbia inserito all'interno dell'immagine $\mathcal{I}$ la marca w. Se qualcun altro da $\mathcal{I}_w$ vuole risalire a w, dopo aver attivato la procedura di recupero di w su $\mathcal{I}_w$, deve rispondere alla domanda: *esiste un w' che equivale a w ?* Questa equivalenza viene valutata attraverso un Test di Somiglianza C che presi in input la coppia (w, w') restituisce 1 se la somiglianza C supera la soglia di somiglianza δ, altrimenti restituisce 0:

$$C(W', W) = \begin{cases} 1 & \text{se } C \geq \delta \\ 0 & \text{altrimenti} . \end{cases}$$

3.1 Un semplice schema di Watermarking

Supponiamo di avere un'immagine $\mathcal{I}$ con n-pixel (Figura 5)

Il possessore dell'immagine $\mathcal{I}$ genera, attraverso un generatore pseudocasuale, una stringa binaria w di n bit, $w = w_1, \ldots, w_n$. Possiamo pensare di organizzare tali n bit in una matrice che, per semplicità di visualizzazione, abbia le stesse dimensione dell'immagine $\mathcal{I}$ (Figura 6).

Consideriamo il pixel generico X_i (una sequenza di bit), l'utente può sostituire l'ultimo bit del pixel X_i con il corrispondente bit w_i (Figura 7).

E' facile osservare che l'immagine così ottenuta non presenta visivamente differenze percettibili da quella originale. L'operazione di rilevamento consiste nella lettura della sequenza degli LSB

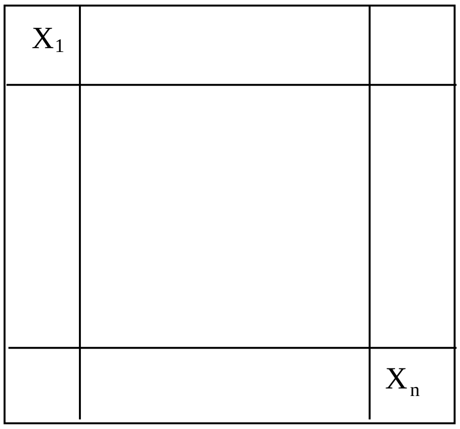

Figura 5: Rappresentazione dell'immagine mediante matrice di pixel

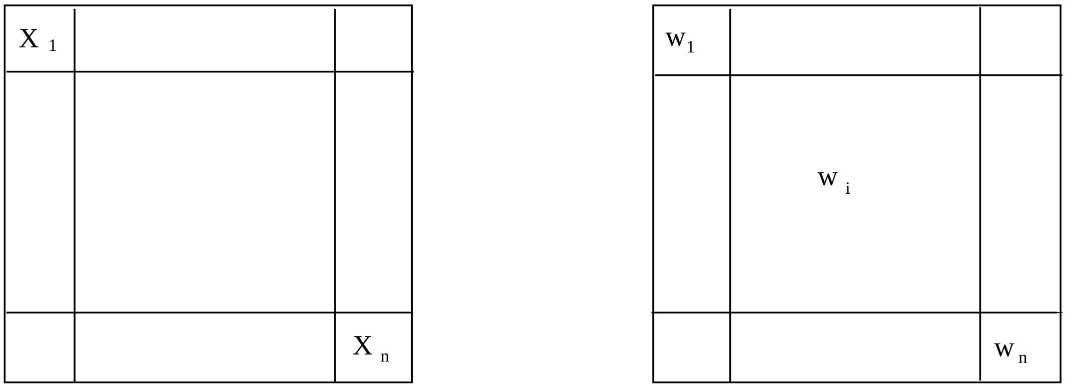

Figura 6: Rappresentazione dell'immagine originale e dell'immagine marcata

(Bit meno significativi) e nel confronto con la stringa generata pseudo-casualmente. Il presente schema è facilmente attaccabile, in quanto data l'immagine marcata basta modificare gli LSB per ottenere un'immagine ancora simile ma priva di watermark. Pertanto, molti dei problemi descritti in principio in questo schema non sono gestibili.

4 Formati grafici

La quantità di bit richiesta per rappresentare un' immagine è elevata. Pertanto, vari formati grafici in cui le immagini vengono memorizzate utilizzano forme di *compressione*. Alcuni comprimono l'immagine originale senza indurre perdite di informazione cioè, una volta decompressa, questa è identica all'originale. Altri utilizzano forme di compressione che inducono perdita di informazione. Il formato GIF appartiene ai formati del primo tipo, mentre JPEG appartiene al secondo.

4.1 JPEG

L'algoritmo di compressione JPEG è uno degli algoritmi più utilizzati per comprimere immagini. Il suo nome deriva dal gruppo di lavoro che lo ha prodotto ovvero da Joint Photographic Experts Group (Gruppo Congiunto di Esperti per la Fotografia). La sua enorme diffusione nelle applicazioni sopra indicate ha suggerito al comitato tecnico che definisce gli standard

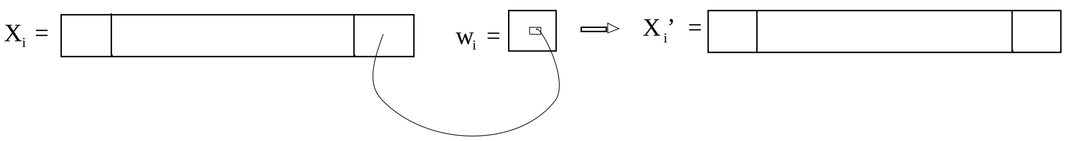

Figura 7: Inserimento della marca

ufficiali per gli algoritmi più usati, composto congiuntamente dallo ISO (International Standards Organization) e dallo IEC (International Electrotechnical Commission), di definirne una versione come standard internazionale [10].
Alla base dell'algoritmo JPEG (ed alla base di tutti gli algoritmi di compressione di immagini) c'è la constatazione che la rappresentazione di un'immagine pixel per pixel contiene informazione ridondante. Per esempio, se metà di un'immagine ha il cielo azzurro, non ha molto senso memorizzare 50.000 volte l'informazione pixel azzurro, pixel azzurro ... etc. La ridondanza, ovvero l'informazione non necessaria e ripetitiva, può essere classificata in vari tipi.

- **Spaziale**: pixel vicini hanno valori simili
- **Percettiva**: il sistema visivo umano è più sensibile ai forti contrasti ed alle variazioni in zone relativamente grandi dell'immagine.

L'algoritmo JPEG sfrutta entrambe le forme di ridondanza. La compressione *elimina i dettagli trascurabili* dal punto di vista percettivo e *raccoglie* le similitudini spaziali. In tal modo la rappresentazione dell'immagine che viene memorizzata risulta più compatta di quella pixel per pixel. D'altra parte, quando l'immagine viene decompressa e visualizzata essa risulta molto simile se non indistinguibile dall'immagine originaria.

4.2 Algoritmo di compressione JPEG

Passi dell'algoritmo per immagini in toni di grigio

- **Partizionamento:** L'immagine originale viene partizionata in blocchi $\mathcal{B}_i$ di dimensione 8×8. Ogni blocco viene trattato come una sottomatrice di valori interi (Figura 8).

- **Rappresentazione nello spazio delle frequenze:** Ad ogni blocco viene applicata la trasformata DCT (Discrete Cosine Trasformation). Precisamente, tale operatore prende in input il blocco $\mathcal{B}_i$ di 8×8 valori e restituisce in output 64 coefficienti appartenenti all'insieme dello spazio delle frequenze. Il primo di questi è detto coefficiente DC mentre i restanti 63 sono detti coefficienti AC.
  I coefficienti cosí ottenuti vengono disposti in ordine *a zig-zag* a partire dall'angolo superiore sinistro andando verso l'angolo inferiore destro. I coefficienti che occupano il triangolo superiore sono quelli più significativi, cioè quelli che hanno valore maggiore e contengono più informazione su $\mathcal{B}_i$. La rappresentazione che si ottiene a seguito dell' applicazione della DCT a tutti i blocchi viene detta rappresentazione nel dominio delle frequenze.

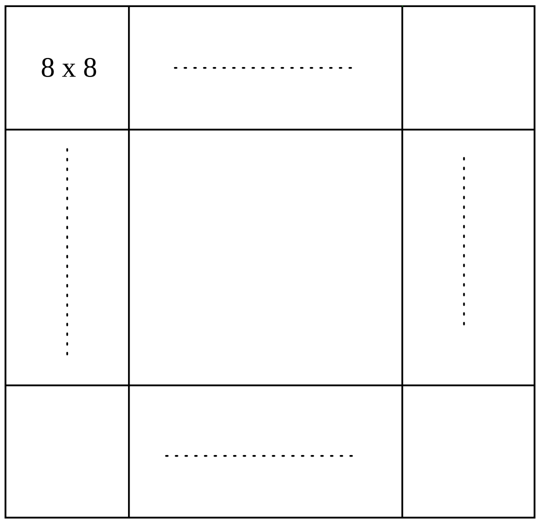

Figura 8: Partizionamento dell'immagine in blocchi

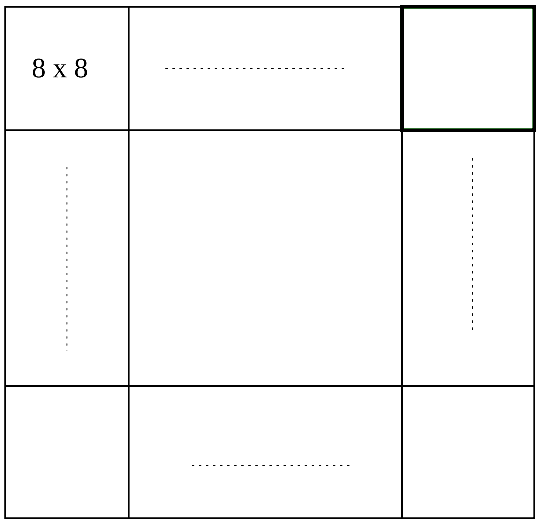

Figura 9: Matrice dei coefficienti

- **Quantizzazione:** I valori ottenuti applicando la DCT vengono quantizzati. Ciò significa che il range di valori di output dalla DCT viene diviso in k sottointervalli e tutti i valori che cadono nel sottointervallo i-esimo vengono sostituiti con un unico valore. L'ampiezza dei sottointervalli viene detta passo di quantizzazione [1]. Per esempio, siano i valori restituiti dalla trasformata DCT nell'intervallo $(0, 2047)$ e supponiamo di avere soltanto 8 bit per rappresentare questi 2048 valori. Dato che con 8 bit si possono rappresentare solo 256 valori, allora si può pensare di suddividere l'intervallo in esattamente 256 sottointervalli, in modo che tutti i valori che cadono nei sottointervalli di $(0, 2047)$ vengono quantizzati con il valore corrispondente ad una delle 256 parti, in cui l'intervallo è stato suddiviso (Figura 10).

- **Codifica entropica:** Ai valori quantizzati viene applicato l'algoritmo di compressione di Huffman. La codifica così ottenuta costituisce la rappresentazione finale dell'immagine

[1] Questo passo induce perdita d'informazione nella procedura di compressione

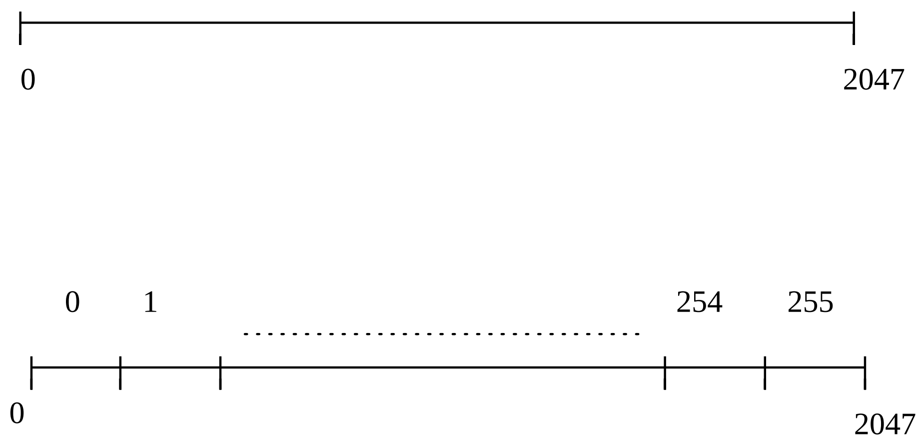

Figura 10: Quantizzazione

che viene memorizzata.

Il passo chiave nell'algoritmo JPEG consiste nell'applicazione della trasformata DCT. Essa costituisce lo strumento matematico che cerca di sfruttare la ridondanza spaziale e percettiva nell'immagine originale. L'idea è la seguente: un'immagine è una funzione di due variabili $f(x, y)$ che appare in forma discreta nella rappresentazione digitale. Cioè i valori dei pixel sono i valori di $f(x, y)$. La funzione $f(x, y)$ può essere approssimata attraverso una somma di funzioni coseno. La trasformata DCT permette di *calcolare esattamente i coefficienti* che compaiono nell'espressione di $f(x, y)$ attraverso le suddette funzioni. Queste funzioni sono una base del nuovo spazio in cui $f(x, y)$ viene rappresentata e ogni coefficiente è una misura della *somiglianza* di $f(x, y)$ alla funzione coseno a cui è associato. L'applicazione di questa trasformazione conviene poichè molti dei coefficienti prodotti dalla DCT sono nulli o vengono poi annullati dalla quantizzazione.

La decompressione dell'immagine in formato JPEG avviene applicando i passi inversi e nell'ordine inverso a quello della procedura di compressione: decodifica di Huffman, una sorta di quantizzazione inversa e l'inversa della trasformata DCT, ossia DCT^{-1}. In realtà, questo schema presenta una perdita di informazione, in quanto la quantizzazione non può essere invertita. Una volta che si è stabilito un intervallo, ad esempio (1,10) e che tutti i valori appartenenti a questo intervallo sono rappresentati da 5, non si sa quali dei valori dell'intervallo considerato è realmente quello corrispondente a 5.

5 Classificazione di schemi di Watermarking.

Gli schemi di watermarking possono essere classificati in :

- **Fragili:** implicano il dannegiamento della marca inserita a seguito di elaborazioni del documento. Tali schemi sono utili quando si vuole utilizzare la marca per controllare eventuali modifiche non autorizzate e quindi verificare l'integrità del documento;

- **Robusti:** il watermark inserito attraverso uno schema robusto è in grado di resistere alle più comuni forme di elaborazione "lecite" del documento cosí marcato. E' resistente anche alle elaborazioni illecite che non rendono il documento inutilizzabile, quindi ad alcuni tipi di attacchi.

Inoltre distinguiamo ancora:

- **Schemi Privati:** richiedono il documento originale in fase di verifica del watermark inserito;
- **Schemi Pubblici:** non richiedono il documento originale.

Una ulteriore classificazione degli schemi:

- **a chiave segreta:** se utilizzano la stessa chiave sia per l'inserimento che per il rilevamento;
- **a chiave pubblica:** se utilizzano una chiave per l'inserimento e una per la verifica, che può essere resa pubblica. In questo caso solo il possessore del documento può inserire il watermark, ma tutti ne possono verificare la presenza.

5.1 Esempi di schemi robusti.

Uno schema per immagini robusto deve essere in grado almeno di *sopravvivere* alle usuali operazioni di image-processing che gli applicativi commerciali rendono possibili: filtering, requantization, dithering, scaling, cropping, resizing, rota-antra, tiling, lossyless compression, lossy compression. Questo è un insieme parziale di possibili trasformazioni che solitamente si effettuano sulle immagini.

SCHEMA 1: L'inserimento richiede la divisione a caso dell'insieme dei pixel dell'immagine di partenza in due sottoinsiemi, A e B, approssimativamente della stessa cardinalità.
Successivamente, scelto un piccolo intero k, occorre aggiungere k al valore dell'intensità di ogni pixel di A e sottrarlo ad ogni pixel di B. Se k è piccolo abbastanza, non si produce nessun degeneramento visibile all'immagine originaria. Ad esempio, supponiamo di avere un'immagine in toni di grigio, in cui ad ogni pixel è associata una stringa binaria di otto bit il cui valore rappresenta un puntatore a uno delle 256 *intensità* di grigio appartenenti alla matrice dei toni di grigio (Figura 11)

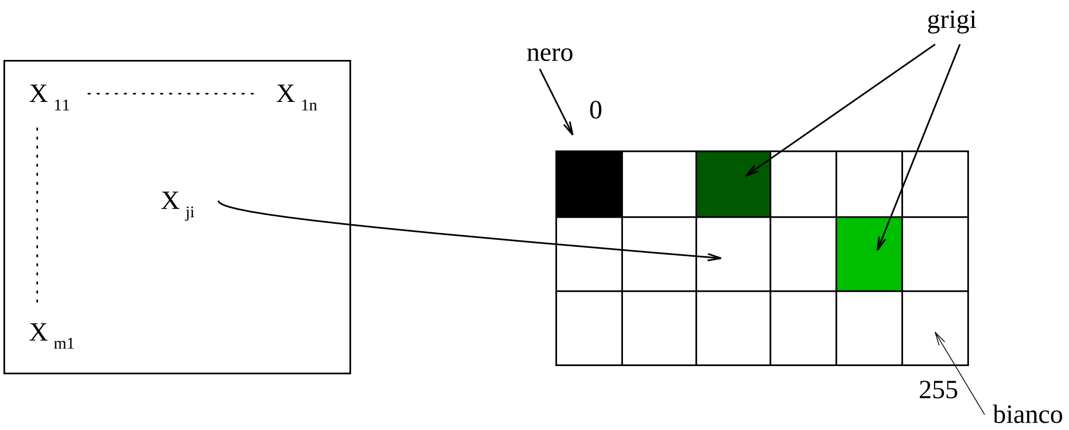

Figura 11: Puntatore alla matrice dei toni di grigio

L'algoritmo per l'inserimento della marca su un'immagine di questo tipo funziona in questo modo:

- si sceglie l'intero k, piccolo (Watermark);
- si divide l'immagine in due sottoinsiemi di pixel A e B, dove $|A| \simeq |B|$ (Figura 12).

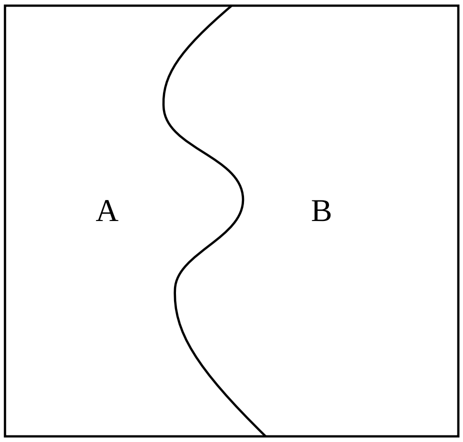

Figura 12: Suddivisione dell'immagine

Supponiamo che tutti i livelli di intensità di grigio *siano equiprobabili*, per cui sulla tabella dei toni di grigio c'è una distribuzione uniforme.
Sia x il valore del puntatore alla matrice di toni di grigio, derivato dall'interpretazione della stringa binaria associata a quel pixel. Indichiamo con $x \in A$ il valore x associato ad un pixel posto in A, con $x \in B$ il valore x associato ad un pixel posto in B, è possibile affermare:

$$\begin{cases} x \in A \text{ allora } x \text{ viene incrementato di } k \\ x \in B \text{ allora } x \text{ viene decrementato di } k. \end{cases}$$

Quindi, a partire dall'immagine $\mathcal{I}$, se ne costruisce un'altra $\mathcal{I}'$, in cui i pixel appartenenti al sottoinsieme A hanno una luminosità maggiore, invece quelli appartenenti al sottoinsieme B hanno una luminosità minore.
Supponiamo che l'utente $\mathcal{C}$, a partire da $\mathcal{I}$, abbia costruito in questo modo $\mathcal{I}'$, e l'abbia resa pubblica. L'utente $\mathcal{D}$ recupera $\mathcal{I}'$ e se ne appropria. Come fa l'utente $\mathcal{C}$ a dimostrare che $\mathcal{I}'$ è una sua creazione? A partire da $\mathcal{I}'$, l'utente $\mathcal{C}$ effettua le operazioni inverse, cioè divide $\mathcal{I}'$ nei due sottoinsiemi A e B (Figura 13)(soltanto l'utente $\mathcal{C}$, che ha trasformato in questo modo l'immagine, conosce la suddivisione precisa in A e B)

e calcola una differenza di medie:

$$\frac{\sum_{x \in A} x}{|A|} - \frac{\sum_{x \in B} x}{|B|}.$$

Ora, nell'immagine di partenza $\mathcal{I}$, una volta effettuata la suddivisione in A e B, se si va a calcolare tale differenza di medie si ottiene un risultato prossimo a zero, in quanto tutti i livelli

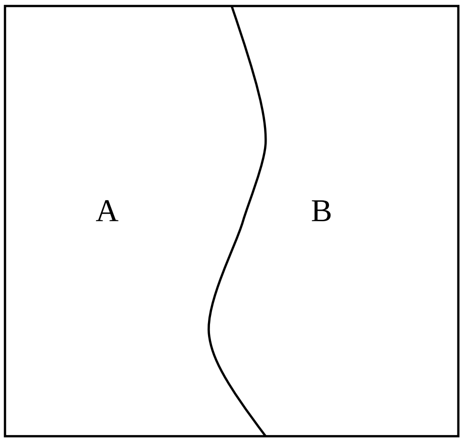

Figura 13: Immagine modificata

di intensità del grigio sono equiprobabili (visto che la $|A| \simeq |B|$). Invece in $\mathcal{I}'$, questa differenza di medie è prossima a $2k$, in quanto i valori del sottoinsieme A sono stati incrementati di k, mentre quelli del sottoinsieme B sono stati decrementati di k, per cui si ha $k - (-k) = 2k$. Ossia nell'immagine originale le due medie coincidono, invece nell'immagine marcata le due medie si allontanano e la distanza è esattamente $2k$; perciò $\mathcal{C}$ può dire: "l'immagine originale è $\mathcal{I}$, l'ho suddivisa in due sottoinsiemi di pixel A e B, ho scelto questo valore k come watermark e su questo valore ho costruito l'immagine $\mathcal{I}'$". $\mathcal{C}$ dimostra cosí di essere il creatore di $\mathcal{I}'$ poichè è l'unico a conoscere il watermark k. Tale schema è robusto rispetto ad alcune trasformazioni.

5.2 Schema per immagini codificate in formato JPEG

Un algoritmo pensato per resistere alla compressione apportata da JPEG funziona in questo modo:

- Data un'immagine I di partenza, si scelgono a caso alcuni blocchi;
- si effettua la trasformazione DCT;
- si considerano nella rappresentazione dello spazio delle frequenze i blocchi corrispondenti a quelli scelti. Ogni blocco di 64 valori in I corrisponde ad un blocco di 64 valori in I' (Figura 14);
- si effettua l'operazione di quantizzazione, ottenendo così una matrice di valori approssimati con coefficienti uniformati.

  L'inserimento della marca avviene modificando i coefficienti a livello DCT ovvero si modificano i coefficienti che corrispondono ai blocchi scelti a caso. Vediamo come avviene tale modifica.
  Sia w la stringa binaria che si vuole inserire all'interno dell'immagine I. Viene preso il primo blocco, che corrisponde al primo bit di w ($= w_1$) scelto casualmente, e si considerano i 64 coefficienti del primo blocco, per cui:

- se il primo bit di w è uno zero, allora si modifica il primo coefficiente (tra i 64) del primo blocco e la modifica viene fatta in modo tale che rappresenti uno zero;

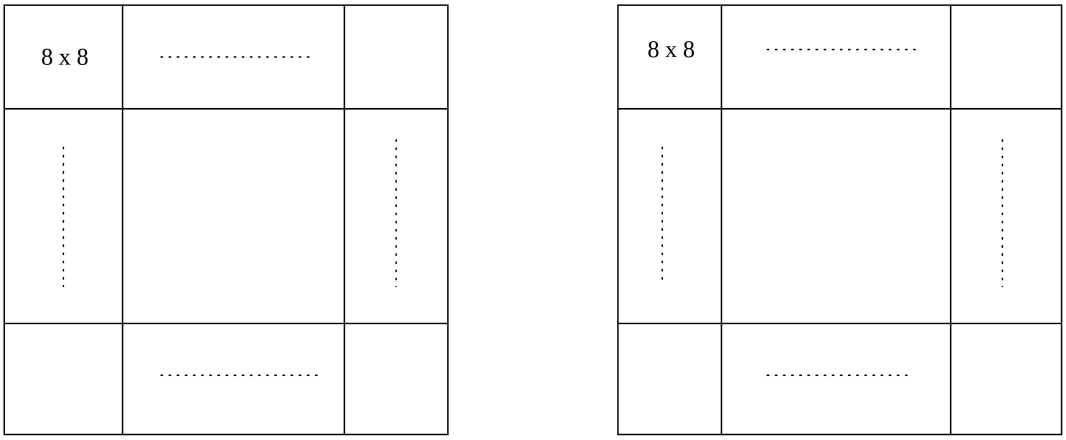

Figura 14: Matrice prima e dopo la DCT

- se si vuole inserire come primo bit di w un uno, allora si modifica il primo coefficiente in modo tale che rappresenti un uno.

Per cui i coefficienti vengono conformati a due relazioni che rappresentano rispettivamente uno zero ed un uno. Il procedimento viene continuato sino a quando w viene esaurita.
In realtà può succedere che alcune modifiche risultino essere troppo pesanti, nel senso che i coefficienti DCT vengono modificati troppo, per cui l'immagine risulta scadente. Quindi, se la modifica è troppo pesante il coefficiente viene marcato come invalido, ciò significa che le relazioni a cui i coefficienti DCT vengono conformati sono complessivamente 3.
Nel momento in cui occorre effettuare la verifica (per esempio la verifica di paternità dell'immagine), si effettua il procedimento inverso: cioè data l'immagine compressa I'_w, si applica un'operazione di decompressione, tramite codici di Huffman, ottenendo i coefficienti DCT dell'immagine I_w. Ottenuti tali coefficienti, a partire dall'originale, è possibile riprodurre la matrice che si ottiene prima dell'inserimento di w e confrontare i coefficienti. Da tale controllo si vede quali sono i coefficienti marcati e a quale relazione sono stati conformati. Quindi si può recuperare la marca w.

5.3 Schema di Cox et al.

Tale schema, dovuto a Cox ed altri [7], è una variazione dell'algoritmo appena presentato. Nello schema menzionato, abbiamo supposto che i blocchi da modificare all'inizio venissero scelti *a caso*. Secondo Cox et al., invece, tali blocchi devono essere scelti in modo intelligente. Una volta applicata la DCT, la matrice risultante ha una struttura ben precisa: ci sono aree in cui sono presenti coefficienti nulli (che non portano informazioni) e aree in cui sono presenti i coefficienti più significativi (quelli che hanno i valori più alti). Occorrerebbe quindi, andare a marcare i valori più significativi in quanto, se si decidesse di marcare i valori nulli, si otterrebbe una perdita di informazione, poichè tali valori vengono tagliati via dagli algoritmi di compressione.

6 Watermarking di immagini: Definizione e Formalizzazione

Prima di descrivere l'algoritmo di Cox et altri, diamo una definizione più formale di problema di Watermark [8]. La Figura 15 illustra il processo di *inserimento* nel quale una marca è inserita nell'immagine e di *recupero* dove la marca è estratta e poi confrontata con quella originaria.

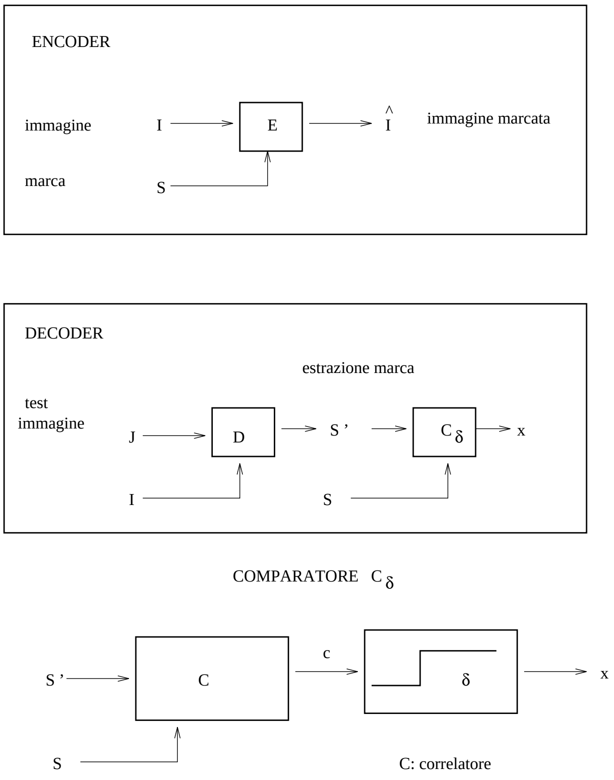

Figura 15: Processo di inserimento e di recupero

Denotiamo con $\mathcal{I}$ l'immagine, con $S = \{S_1, S_2, \ldots\}$ l'insieme delle possibili marche e con $\hat{\mathcal{I}}$ l'immagine marcata. ε è una funzione di encoder se, prendendo un'immagine $\mathcal{I}$ e la marca S, genera una nuova immagine chiamata *immagine marcata*, cioè $\varepsilon(\mathcal{I}, S) = \hat{\mathcal{I}}$. Una funzione di decoder $\mathcal{D}$ prende un'immagine $\mathcal{J}$ elaborata dal possessore ed estrae la marca S' dell'immagine, cioè $D(\mathcal{J}, \mathcal{I}) = S'$. La marca estratta S' sarà confrontata con la sequenza di marche del possessore tramite la funzione di comparazione C_δ che genera un output binario:

$$C_\delta(S', S) = \begin{cases} 1 & \text{se } c \geq \delta \\ 0 & \text{altrimenti} \end{cases}$$

dove c è la correlazione delle due marche.

Senza perdere di generalità uno schema di watermarking può essere visto come una tripla $(\varepsilon, D, C_\delta)$.

La classe di schemi di watermarking che considereremo ora è detta *feature-based*, in quanto include una marca S in alcuni degli elementi di $D(\mathcal{I}) = \{f_1(\mathcal{I}), f_2(\mathcal{I}), \ldots\}$, detto insieme delle caratteristiche dell'immagine I.
Il processo di inclusione è ottenuto tramite una operazione di inserimento denotato con il simbolo $\oplus$, cioè $f_i' = f_i \oplus S_i$.
L'operazione di inserimento ha un'operazione inversa, chiamata operazione di estrazione, denotata con $\ominus$, cioè $f_i' \ominus f_i = S_i$.

6.1 Schema di Cox

In questo schema l'insieme delle caratteristiche $D(\mathcal{I})$ corrisponde agli n coefficienti AC più significativi (di bassa frequenza) ottenuti applicando la DCT all'immagine $\mathcal{I}$. Modifiche rilevanti su tali coefficienti causano il degrado della fedeltà dell'immagine.

Una marca in tale schema è un vettore di valori reali $S = s_1, \ldots, s_n$ casuali, scelti in base ad una distribuzione di tipo gaussiano.

La procedura di inserimento ε prende in input la marca S e l'immerge nell'insieme $D(\mathcal{I})$ attraverso una semplice somma.

A questa matrice di coefficienti modificata viene applicata la DCT^{-1}, ottenendo l'immagine marcata $\mathcal{I}'$ che viene resa pubblica.

Per determinare se una data immagine $\mathcal{J}$ contiene una marca S, la decodifica D, per prima cosa, estrae $T = \{t_1, t_2, \ldots\}$ da $\mathcal{J}$ come segue: $t_i = f_i(\mathcal{J}) - f_i(\mathcal{I})$.
La misura di confidenza C sarà:

$$C = \frac{\Sigma_i t_i s_i}{\sqrt[2]{\Sigma_i t_i^2}}$$

o alternativamente in forma normalizzata:

$$C = \frac{\Sigma_i t_i s_i}{\sqrt[2]{(\Sigma_i t_i^2 \Sigma_i s_i^2)}}.$$

In questo caso se $\mathcal{J} = \mathcal{I}'$, allora $C = 1$. Se $\mathcal{J}$ è una versione modificata di $\mathcal{I}'$, e i cambiamenti non sono percettibilmente significativi, C sarà un valore grande ma < 1.

6.2 Schemi invertibili

Lo schema appena descritto può essere sottoposto al seguente attacco [8]: Supponendo I_w pubblica si può semplicemente scegliere una stringa w', sottrarla da I_w e costruire una nuova immagine $I_w - w' = I'$, e poi dire che I_w si ottiene semplicemente sommando a $\mathcal{I}$ la marca w'. Ma in questo caso, sono state generate due coppie (I, w) e (I', w') che collidono. Se ciò accade il legittimo proprietario dell'immagine non riesce a dimostrare i propri diritti sull'immagine in quanto w non può essere usata come prova. Come nel caso delle funzioni hash, anche per la digital watermarking technology si richiede che sia computazionalmente inammissibile trovare delle collisioni. Formalmente, si richiede il soddisfacimento della seguente proprietà:

Definizione *di schemi di watermark invertibili.*
Sia $\mathcal{V}$ la classe di schemi di watermarking invertibili. Indichiamo con $\mathcal{S}$ una marca, e $\mathcal{S}'$ una

seconda, tale che sia $\mathcal{S}$ che $\mathcal{S}'$ appartengano all' insieme delle marche ammissibili.
Data un'immagine $\mathcal{I}$, lo schema $(\mathcal{E}, \mathcal{D}, \mathcal{C}_\delta) \in \mathcal{V}$ se per ogni $\mathcal{S}$ e $\mathcal{S}'$ esiste un $(\mathcal{E}', \mathcal{D}, \mathcal{C}_\delta)$ con

$$\mathcal{E}'(\hat{I}, \mathcal{S}') = \hat{I}'$$

e

$$\mathcal{E}(\hat{I}', \mathcal{S}') = \hat{I}$$

tale che $\hat{I} = \mathcal{E}(I, \mathcal{S})$ e $\mathcal{E}'$ è una funzione di codifica computazionalmente ammissibile, dove non c'è un calo di qualità percepibile nelle immagini derivate $\hat{I}$ e $\hat{I}'$, da I.
Se ciò non accade $(\mathcal{E}, \mathcal{D}, \mathcal{C}_\delta)$ si dice *non invertibile.*

6.3 Esempi di schemi fragili

L'obiettivo di questi schemi è di tener traccia di ogni tipo di modifica apportata all'immagine ed eventualmente di localizzarla. Seguono le descrizioni di alcuni schemi ritenuti sicuri.

SCHEMA 1: Supponiamo di avere un'immagine I rappresentata da una matrice di pixel in cui ogni pixel è una stringa binaria, e supponiamo di disporre di una stringa casuale R che utilizzeremo come stringa di riferimento (Figura 16) (tale stringa è pubblica).

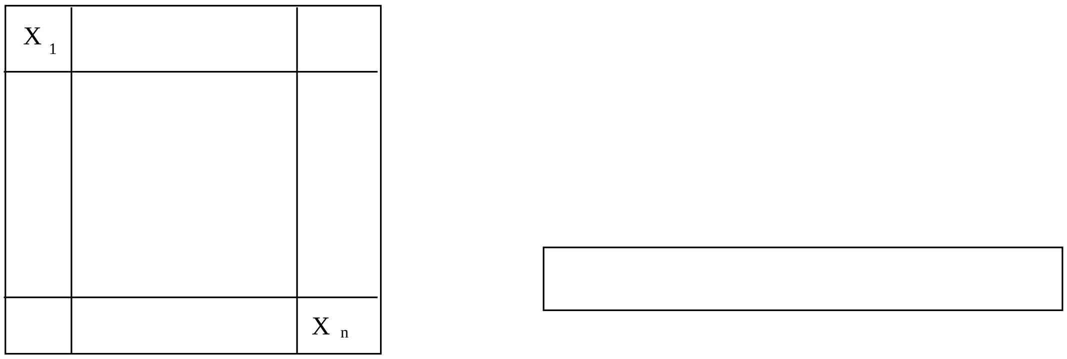

Figura 16: Immagine e stringa di riferimento

Per proteggere la nostra immagine, possiamo generare una stringa binaria $w = w_1, \ldots, w_n$ da inserire nell'immagine I. A questo punto scorriamo l'immagine dal primo all'ultimo pixel, poichè ogni pixel è una stringa di bit, possiamo considerarla come un puntatore ad una posizione della stringa R (Figura 17).

Se il primo bit di w, che vogliamo inserire all'interno dell'immagine I, è zero, allora andiamo a vedere la rappresentazione del pixel X_1 a quale bit di R punta. Se il bit puntato è uno zero, allora lasciamo X_1 cosí com'è, senza modificarlo. Supponiamo che w_2=1, allora andiamo a vedere il pixel X_2 a quale bit di R punta. Se tale bit è un 1, allora lasciamo X_2 cosí com'è, se invece è uno zero allora andiamo a modificare leggermente X_2 (per esempio, modifichiamo il suo ultimo bit), in modo che vada a puntare ad un altro bit di R. Se il nuovo valore puntato è un 1,allora lasciamo X_2 come l'abbiamo modificato; se invece è ancora uno zero, effettuiamo un'altra modifica (sempre sugli ultimi bit) e continuiamo fino a quando non troviamo in R un

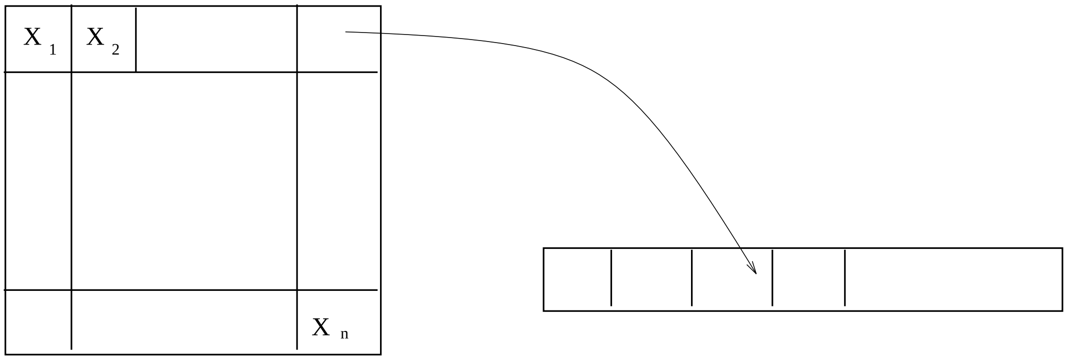

Figura 17: Utilizzo della stringa di riferimento

bit uguale a 1. Nel momento in cui troviamo un bit che "combacia" con il bit che vogliamo inserire, arrestiamo il nostro procedimento. Stiamo utilizzando la stringa R come una stringa per riferimento, mediante cui modifichiamo i nostri pixel, in modo che, visti come puntatori alla stringa R, individuino la stringa w, che abbiamo scelto come marca. Poichè la stringa R è casuale, la probabilità di trovare un 1 o uno 0 è 1/2; per cui, dopo al più due tentativi, in media, riusciamo a trovare un bit che "combacia" con quello che vogliamo inserire. In fase di verifica, avendo w, essendo R pubblica e leggendo i pixel dal primo all'ultimo, si riesce a vedere il pixel al quale il bit punta e a costruire una stringa binaria(per esempio 0101...).Se tale stringa risulta essere uguale a w, allora l'immagine resa pubblica non è stata modificata. Nel momento in cui, invece, si riscontra una differenza tra le due stringhe, siamo anche in grado di risalire a quale pixel è stato modificato.
Uno schema più sofisticato fa invece uso delle funzioni hash.

SCHEMA 2: Supponiamo di avere un'immagine I, rappresentata da una matrice di pixel, che dividiamo in blocchi di dimensione 8*8 (ossia di 64-pixel) (Figura 18).

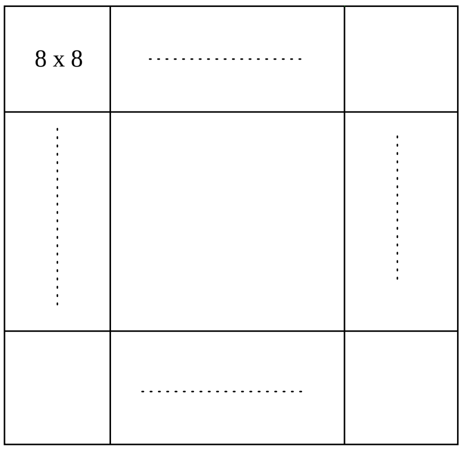

Figura 18: Suddivisione dell'immagine in blocchi

Si effettuano le seguenti operazioni:

- vengono rimossi da ogni blocco gli LSB di tutti i pixel;

- vengono concatenati i bit più significativi di ogni pixel componendo un'unica grossa stringa. Quando effettuo la concatenazione, posso aggiungere ad essa anche delle informazioni tipo una chiave segreta k, parametri dell'immagine, ecc.;

- si effettua l'hash di tale stringa;

- si effettua un'operazione di XOR tra l'hash calcolato e una stringa w, che è il watermark;

- si inserisce la stringa risultante al posto dei bit meno significativi, che erano stati rimossi.

- Questo procedimento viene ripetuto per ogni blocco.

Alla fine ciò che ottengo è un'immagine che si differenzia dall'originale soltanto nei bit meno significativi, ma la costruzione è stata fatta in modo che tali bit meno significativi siano legati a tutta l'immagine I. In fase di verifica si può facilmente dimostrare che,presa l'immagine originale I, la stringa w è stata inserita con tale procedimento.

Esiste una versione dello schema descritto che utilizza la crittografia a chiave pubblica.

SCHEMA 3: L'immagine originaria viene partizionata in blocchi come in precedenza, gli LSB vengono eliminati. I rimanenti MSB (Bit più significativi) vengono concatenati con una stringa binaria che rappresenta i parametri dell'immagine. Alla stringa complessiva, viene applicata una funzione hash. Quindi, il message digest prodotto viene cifrato attraverso la chiave privata. Del messaggio cifrato cosí ottenuto viene fatto l'XOR con il watermark w e la stringa risultante viene inserita negli LSB dei blocchi dell'immagine.
L'estrazione richiede il processo inverso. Prima di tutto occorre concatenare i MSB di ogni blocco con i parametri dell'immagine e farne l'hash. Successivamente, bisogna decifrare attraverso la chiave pubblica la stringa degli LSB dei vari blocchi. Infine facendo l'XOR del message digest prodotto con la stringa decifrata si riottiene il watermark w.

6.4 Immagini in bianco e nero

Per queste immagini si possono produrre schemi di watermarking che utilizzano la *crittografia visuale*. Con uno schema a soglia è possibile inserire in un'area dell'immagine apparentemente priva di significato un watermark che "compare" soltanto quando due o più immagini vengono sovrapposte.

7 Caratteristiche degli schemi

La tecnologia del digital watermarking è uno strumento che permette una più ampia *condivisione* delle informazioni garantendo protezione nei confronti dei pirati.
Il ciclo di vita di un watermark consta di quattro passi: inserimento, distribuzione del documento, rilevamento e interpretazione. I soggetti che entrano in gioco in uno schema di watermark sono tre: proprietari dei documenti, clienti e attaccanti. Naturalmente i desideri di queste tre figure sono radicalmente diversi.
I proprietari dei documenti sono solitamente gli acquirenti del software per l'inserimento delle marche nei propri prodotti. Il loro interesse principale è quello di proteggere i documenti. Poichè solitamente l'inserzione di un watermark comporta una perdita in qualità del documento, essi debbono scegliere tra il livello di protezione desiderato e la degradazione che viene

indotta con l'inserimento del watermark. E' auspicabile che gli schemi risultino *parametrizzabili* per essere adattabili a diversi casi.

Quando il documento viene distribuito lo schema deve stabilire quali sono le modalità di accesso al watermark. Alcuni schemi possono prevedere che soltanto il proprietario del documento sia in grado di recuperare il watermark inserito, mentre altri possono fornire al cliente il software di rilevamento.

Il rilevamento del watermark può o meno richiedere il documento originale. Inoltre può richiedere una forma di preprocessing del documento.

L'interpretazione del watermark rilevato da un'immagine marcata è un passo di fondamentale importanza. Gli schemi, per essere utilizzati in sedi giuridiche, debbono prevedere una interpretazione chiara e non ambigua della stringa rinvenuta nel documento.

Gli acquirenti dei documenti desiderano che la presenza del watermark sia impercettibile: la qualità del loro documento deve essere identica o quasi a quella dell'originale e le modalità d'uso dei dati non debbono risentire in alcun modo della presenza della marca. Per esempio, un watermark immerso in un video digitale potrebbe comunicare al device di lettura l'autenticità o meno della copia introdotta. In tal modo le copie contraffatte dovrebbero essere riconosciute dal device e non riprodotte sullo schermo. Naturalmente un utente legale non ammetterebbe di non poter veder la propria copia perchè il device nel leggere la marca ha commesso un errore di valutazione! Quindi gli schemi di watermark debbono essere "robusti" anche in questo senso più generale.

Lo scopo principale di un pirata è quello di "liberare" il documento dalla presenza del watermark. Esistono diversi tipi di attacchi che possono raggiungere il suddetto scopo: attacchi contro la *robustezza del watermark*, attacchi di *presentazione*, attacchi di *interpretazione* ed attacchi *legali*.
Gli attacchi di robustezza mirano alla rimozione diretta del watermark senza danneggiare il documento oltre misura. A tal fine possono essere utilizzate sia trasformazioni lecite che operazioni illecite.
Quelli di presentazione mirano a far in modo che il device addetto al riconoscimento del watermark non riesca nel compito. Solitamente questi attacchi mirano a "presentare" il documento al device in una forma disorientante.
Gli attacchi di interpretazione hanno invece il fine di non rendere chiara la presenza del watermark. Essi non mirano alla rimozione ma, spesso, all'aggiunta di altri watermark che entrino in conflitto con l'originale e rendano impossibile arrivare a stabilire qual è quello autentico.
Gli attacchi legali sono di vario tipo. Alcuni possono sfruttare le debolezze note degli schemi per giungere ad affermarne l'inattendibilità, mentre altri possono prescindere totalmente dalla natura tecnica e puntare sulle falle della legislazione vigente.

8 Standardizzazione

Le applicazioni che possono beneficiare degli schemi di watermarking possono essere raggruppate in tre cluster che richiedono tecniche simili: applicazioni che usano i watermark per trasmettere informazioni sui *diritti di proprietà* dei dati, applicazioni che usano i watermark per *verificare integrità ed autenticità* dei dati e applicazioni che utilizzano i watermarks per *inviare informazioni* ad una comunità di riceventi.

Applicazioni che appartengono ai primi due cluster sono facili da individuare. Due interessanti applicazioni appartenenti al terzo cluster sono *il monitoraggio di un sistema audio*

e la protezione di *video digitali*. Nel primo caso possiamo considerare il seguente scenario: un'unità trasmittente invia in broadcast brani sonori. I vari pezzi di un file audio sono marcati attraverso identificatori unici. Un dispositivo di monitoraggio permanente controlla il passaggio di tutti i pacchetti, estrae le marche e automaticamente effettua il conteggio dei pacchetti inviati ai vari utenti per i pagamenti da richiedere a chi effettua il broadcast.
Nel secondo lo scenario da considerare è quello del video digitale. Con l'introduzione del DVD (Digital Versatile Disc) è diventato pressante il problema di individuare sistemi che permettano di evitare le copie illegali. Le marche possono essere inserite nelle immagini video al fine di comunicare al device che effettua la riproduzione informazioni sull'autemticità della copia, sull'integrità, sul numero di copie ammesse e cosí via, a seconda del tipo di utente che acquista il prodotto.

La tecnologia dei watermark può essere paragonata per certi aspetti a quella della *compressione*. Il ricevente deve possedere una descrizione dettagliata delle modalità attraverso le quali i dati vengono compressi dal distributore di informazioni. Come esistono molteplici tecniche per la compressione dei dati cosí esistono molte proposte di schemi di watermarking. Nascono allora vari problemi. Occorre stabilire delle regole generali che possano guidare i progettisti di schemi di watermarking affinchè le loro realizzazioni possano essere facilmente integrati in applicazioni più generali. La *standardizzazione delle interfacce*, per esempio, permetterebbe la realizzazione di applicazioni che possono in qualunque momento sostituire lo schema di watermarking qualora ne venisse prodotto uno più confacente all'applicazione. Occorre individuare degli *strumenti di valutazione* che permettano di esprimere un giudizio, per esempio nel caso degli schemi robusti, sulla bontà delle soluzioni proposte. A tal fine un insieme di *benchmark* potrebbe essere di aiuto. Un tale insieme dovrebbe implementare sia gli attacchi realizzabili attraverso le cosiddette trasformazioni lecite sul documento che quelle illecite. Attacchi tipici sulle immagini marcate sono i seguenti:

- *Attacchi con aggiunta di rumore*. Il watermark viene attaccato aggiungendo un valore random ad ogni pixel. I valori devono essere abbastanza piccoli al fine di non degradare pesantemente l'immagine.

- *Attacchi con marcature multiple*. L'attaccante spera di danneggiare il watermark iniziale inserendo nell'immagine un secondo watermark nella stessa immagine.

- *Attacchi interattivi*. Tali attacchi sono apportabili quando è pubblica la procedura di rilevamento del watermark. In tal caso, infatti, è possibile modificare l'immagine, estrarre il watermark per valutare il danno apportato e ripetere l'operazione fino a quando la presenza del watermark non viene eliminata.

- *Attacchi di inversione*. L'attaccante cerca prima di individuare il watermark nel documento. Poi cerca di determinarlo e infine di rimuoverlo. Il processo di rimozione in questo caso è l'inverso di quello di immersione (se il watermark è stato aggiunto attraverso un'addizione, la rimozione avviene tramite una sottrazione...).

- *Attacchi per cospirazione*. Più cospiratori posseggono copie diverse della stessa immagine, che differiscono soltanto perchè ognuna di essa ha un differente watermark. L'attacco consiste nel produrre una nuova immagine i cui pixel sono ottenuti mediando i pixel delle immagini in possesso dei cospiratori ed eventualmente aggiungendo del rumore. L'auspicio è che la nuova immagine dovrebbe essere ancora "vicina" a quelle vendute ai cospiratori ma non dovrebbe presentare più le marche in esse presenti.

9 Conclusioni

I nuovi media che memorizzano l'informazione e le nuove forme di trasmissione pongono una serie di problemi nuovi in merito alla *protezione* dei dati.
La tecnologia dei watermark sembra essere una risposta efficiente alla vasta gamma di questioni che la digitalizzazione dell'informazione ha sollevato. Oltre alle applicazioni descritte in precedenza i watermark possono essere anche visti come uno strumento per *aggiungere valore* ai dati stessi. Un watermark potrebbe "racchiudere" codice eseguibile che, debitamente interpretato dal device a cui i dati vanno in pasto, fornisce determinate funzionalità. I watermark possono essere usati in documenti ipertestuali per fornire "collegamenti occulti" a siti specifici. I gestori di tali siti potrebbero pagare i produttori di pagine web al fine di "dirottare" l'attenzione degli utenti sulle proprie pagine. Inoltre, i watermark potrebbero essere usati come tramite tra i nuovi media e la carta tradizionale. Se, infatti, fossero in grado di resistere ad una conversione analogico-digitale, i diritti di proprietà sui dati si manterrebbero anche dopo aver effettuato, per esempio, una stampa.
Occorre fare due osservazioni: la prima è che il problema della protezione è reso ancora più impellente dal fatto che le copie degli originali digitali sono *esattamente* identiche agli originali. Se le tecnologie tradizionali comportavano comunque una perdita di qualità nell'operazione di copia, si pensi alle copie delle cassette audio, le copie digitali non subiscono alcuna forma di degradazione. La seconda riguarda il valore che i dati digitali hanno.
Non tutti i dati hanno la stessa importanza; pertanto gli schemi di watermarking dovrebbero essere *scalabili*. Dati più importanti dovrebbero essere protetti con un alto grado di sicurezza e a tal fine potrebbero richiedere anche un alto sforzo computazionale, ma dati non importantissimi potrebbero anche avere una protezione limitata nel tempo. Si pensi, per esempio alle diverse richieste di protezione che possono avere le pagine di un quotidiano digitale dal video digitale di un film quale "Via col Vento".
Come si intuisce, quindi, le considerazioni da far presente a coloro che vedono nei watermark una risposta efficiente per risolvere alcuni problemi della digitalizzazione dell'informazione sono ancora molte. La tecnologia del digital watermarking è estremamente recente. I soggetti interessati al suo sviluppo sono molteplici: università, centri di ricerca, piccole e medie imprese. Essa può rappresentare un grosso business e può essere *l'agente abilitante* per molti altri. Affinchè prenda completamente piede, però, è necessario che vengano compiuti sforzi non soltanto nell'ambito tecnologico ma anche in quelli giuridico, economico e sociale. Il valore e il grado di accettazione che tale tecnologia avrà presso gli utenti dipenderà in buona sostanza dagli sviluppi che in parallelo saranno compiuti nella ricerca teorica, negli apparati giuridici e nelle scelte e negli investimenti di mercato.

Riferimenti

[1] M.M. Yeung "Digital Watermarking", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[2] N. Memon e P.W. Wong "Protecting Digital Media Content", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[3] S. Craver, B.L. Yeo e M.M. Yeung "Technical Trials and Legal Tribulations", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[4] F. Mintzer, G.W. Braudaway e A.E. Bell "Opportunities for Watermarking Standards", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[5] J. Zhao, E. Koch e C. Luo "In Business Today and Tomorrow", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[6] J.M. Acken "How Watermarking Adds Value to Digital Content", *Communications of the ACM*, July 1998, vol. 41 no. 7.

[7] I.J. Cox, J. Kilian, T. Leighton e T. Shamon "Secure Spread Spectrum Watermarking for Multimedia", *NEC Research Institute Technical Report*, 95-10.

[8] S. Craver, N. Memon, B.L. Yeo e M.M. Yeung "Can Invisible Watermarks Resolve Rightful Ownerships?", *IBM Research Report*, RC 20509 July 25, 1996.

[9] S. Battista "JPEG E MPEG", *BIT*, Gennaio 1995.

[10] ISO/IEC Is 10918 "Digital Compression and Coding of Continuous-tone Still Images", Settembre 1992.