Università degli Studi di Padova

Dipartimento di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria Informatica

Sicurezza nello standard DICOM: Firma Digitale e Watermark

Laureando:
Mirko Signorato

Relatore:
Ch.mo Prof.**Ennio Buro**

Anno accademico 2009/2010

Indice

Introduzione

La presente relazione si basa su un tirocinio lungo da me svolto della durata di circa sei mesi, cominciato l'8 febbraio 2010 e finito il 30 agosto 2010. Per l'occasione sono stato ospitato dalla Network Solutions for Business (N.S.B.) di Altavilla Vicentina(VI), con la quale ho collaborato alla progettazione e costruzione di una macchina proprietaria (Appliance) che fosse in grado di carpire immagini digitali da un generico radiografico e da queste creasse file in formato DICOM. In quest'esperienza sono stato seguito dal Sig. Marco Zuffolato, co-titolare dell'azienda e mio tutor aziendale. Il mio tutor universitario, nonchè relatore, è il professor Ennio Buro, docente di Ingegneria del Software.

Capitolo 1

Lo standard DICOM

1.1 Introduzione

Lo standard DICOM (Digital Imaging and COmmunications in Medicine) definisce i criteri per la comunicazione, la visualizzazione, l'archiviazione e la stampa di informazioni di tipo biomedico quali ad esempio immagini radiologiche. Spesso si tende a pensare che lo Standard DICOM sia semplicemente un formato d'immagine o di file; questa definizione è assolutamente errata, in quanto è stato pensato per coprire tutti gli aspetti funzionali delle immagini digitali in campo medico.

Occorre notare che il DICOM è uno standard industriale, e non uno standard ISO, quindi universale: ciò comporta una certa tolleranza nell'implementazione delle specifiche, al punto che attualmente forse non esistono apparecchiature che possano definirsi pienamente DICOM compliant (compatibili), nel senso rigoroso che la definizione di uno standard imporrebbe. Nella maggior parte dei casi, infatti, un'apparecchiatura risulta conforme ad una parte dello standard (ad esempio la modalità di archiviazione delle immagini), mentre adotta tecnologie proprietarie per altre funzionalità (ad esempio la gestione delle liste pazienti).

Lo standard DICOM è pubblico, nel senso che la sua definizione è accessibile a tutti. Tutta la documentazione si trova nel sito della NEMA (medical.nema.org) e consta di 18 parti, ognuna delle quali definisce un preciso aspetto del DICOM. La sua diffusione si rivela estremamente vantaggiosa perché consente di avere una solida base di interscambio di informazioni tra apparecchiature di diversi produttori, raggiungendo così l'obiettivo di rendere indipendenti le immagini radiografiche dallo specifico produttore. Nel campo ospedaliero pubblico è già diffusamente adottato, sia a livello nazionale che a livello internazionale. Per quanto riguarda il campo privato (e in particolar modo l'odontoiatria) è ancora ignorato, rendendo così estremamente difficile la comunicazione di dati e immagini del paziente tra diversi studi medici privati.

1.2 Storia del DICOM

Il progetto originario venne sviluppato da due associazioni statunitensi: The American College of Radiology (ACR), responsabile dello sviluppo tecnico-medico del sistema, e il National Electrical Manufacturers Association (NEMA). Nel 1985 venne ufficializzata la versione 1.0 dello standard DICOM a cui seguì nel 1988 la versione 2.0: si trattava di un primitivo standard in cui era definito il formato dei file contenenti le immagini e lo standard fisico e di protocollo per l'interconnessione punto-punto delle varie apparecchiature. Le implementazioni tuttavia furono piuttosto limitate, soprattutto a causa del mezzo fisico di connessione realizzato con tecnologie già per l'epoca obsolete. Nel 1993 lo standard DICOM si trasformò radicalmente nella versione 3.0 nella quale, mantenendo sostanzialmente immutate le specifiche inerenti il formato delle immagini, furono aggiunti numerosi servizi ed implementati i protocolli di rete TCP/IP e OSI: il nuovo standard venne identificato con il termine DICOM e proprio l'integrazione nelle specifiche del protocollo di rete TCP/IP, ormai largamente diffuso, ne decretò un successo ed una popolarità sempre crescenti. Ad oggi, nel campo radiologico digitale, è certamente lo standard internazionale più utilizzato. Attualmente, gli sviluppatori del progetto pubblicano annualmente una revisione dello standard, con la quale il DICOM resta aggiornato sui sviluppi tecnologici della medicina.

Breve riepilogo:

- 1985: Standard DICOM 1.0
- 1988: Standard DICOM 2.0
- 1993: Standard DICOM 3.0

1.3 Il file DICOM

Un file DICOM nella sua essenza può essere visto come un contenitore; questo standard non introduce nessun nuovo formato per le immagini (come i tipi jpeg, gif ecc.). I dati contenuti non rappresentano solamente i pixel e i colori dell'immagine, ma anche un'insieme strutturato di dati che descrivono tutto il procedimento che ha portato alla costruzione dell'immagine stessa. Sostanzialmente un file DICOM consiste quindi di un'intestazione (header) costituita da un insieme di attributi contenenti informazioni di varia natura e da un corpo dati atto a contenere una o più immagini. I dati immagine possono essere anche compressi attraverso una ampia varietà di algoritmi standard di compressione immagini, tra i quali JPEG, JPEG2000 LZW and Run-length encoding (RLE). L'insieme di attributi che formano l'intestazione possono essere raggruppati in base alle relazioni che esistono tra di loro e vanno a formare un'entità conosciuta come

oggetto informativo. Ciascun gruppo può quindi rappresentare l'astrazione di un'entità reale, quali ad esempio: Paziente, Studio, Serie e Immagine. Nel primo gruppo di attributi sono presenti informazioni generali sul paziente sottoposto a indagine medica(nome, ID, data di nascita, sesso, ...); nel secondo gruppo sono presenti le caratteristiche delle diverse metodiche di analisi costituenti lo studio diagnostico (data, ora, medico referente, ...); nel terzo gruppo, definito serie, vengono raccolti i dati che descrivono le collezioni di immagini con i relativi parametri di acquisizione (numero della serie, tipo di madalità, ...); infine il quarto è ultimo gruppo contiene gli attributi descrittivi delle immagini come la dimensione della matrice, la profondità del pixel, l'interpretazione fotometrica, ... Naturalmente questi non sono i soli gruppi presenti, ma sono certamente fondamentali ai fini di una descizione completa della radiografia. Come detto, lo standard DICOM differisce da altri formati per la sua capacità di portare con sè informazioni riguardanti tutta la storia clinica della radiografia. Possiamo quindi equiparare un file DICOM ad una cartella clinica. Strutturalmente un attributo contenuto nell'header è formato da un insieme di campi:

- un'etichetta (tag): è composto da due numeri esadecimali a 4 cifre, uno per il campo group e uno per il campo element; il primo rappresenta il gruppo di appartenenza dell'attributo (Paziente, Studio, ...) e l'altro rappresenta l'elemnto specifico all'interno del gruppo;

- un nome: è un indicatore mnemonico per facilitare la leggibilità degli attributi;

- una lunghezza: indica la lunghezza in byte del valore dell'attributo

- il tipo di dato (VR): nel prossimo sottoparagrafo sono elencati i 27 possibili tipi di dato;

- il valore dell'attributo;

Nello standard sono previsti all'incirca 2000 attributi predefiniti, alcuni dei quali obbligatori. Il DICOM permette inoltre la creazione di altri attributi, detti privati, i quali possono aggiungere ulteriore informazione alla radiografia. É doveroso precisare che i pixel stessi dell'immagine sono contenuti in un paticolare attributo, chiamato *Pixel Data*, il quale può essere di 3 tipi diversi: OB, OW e OF, rispettivamente per le immagini a 8 bit per pixel, 16 bit per pixel o 32 bit per pixel.

Elenco dei 27 tipi di dato

VR	**Definizione**
AE	Application Entity
AS	Age String
AT	Attribute Tag
CS	Code String
DA	Date
DL	Delimitation
DS	Decimal String
FL	Floating Point Single
FD	Floating Point Double
IS	Integer String
LO	Long String
LT	Long Text
OB	Other Byte String
OF	Other Float String
OW	Other Word String
PN	Person Name
SH	Short String
SL	Signed Long
SQ	Sequence of Items
SS	Signed Short
ST	Short Text
TM	Time
UI	Unique Identifier
UL	Unsigned Long
UN	Unknown
US	Unsigned Short
UT	Unlimited Text

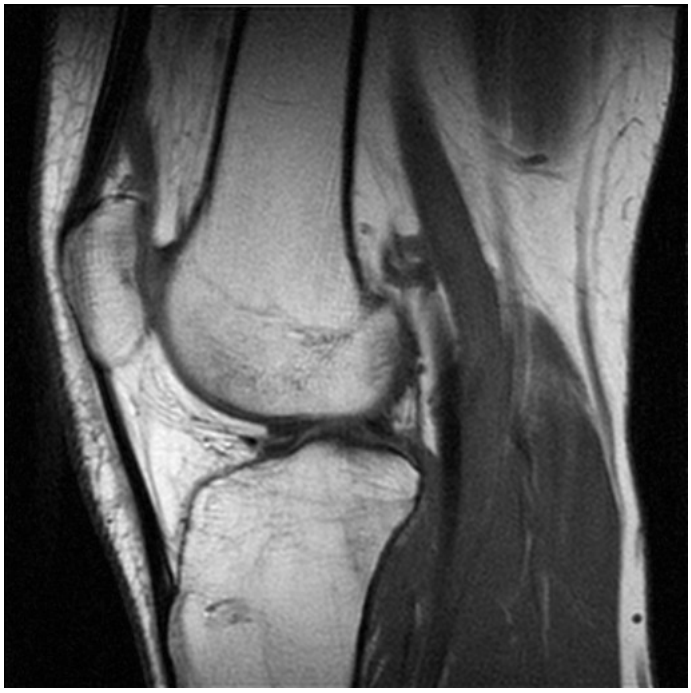

Figura 1.1: Esempio d'immagine

Group - Element	Description	Type	Length	Value
0002 0000	Group 0002 Length	UL	4	200
0002 0001	File Meta Information Version	OB	2	(binary data)
0002 0002	Media Storage SOP Class UID	UI	26	1.2.840.10008.5.1.4.1.1.7
0002 0003	Media Storage SOP Instance UID	UI	54	1.2.826.0.1.3680043.2.1208.34948545565201041914916296
0002 0010	Transfer Syntax UID	UI	18	1.2.840.10008.1.2
0002 0012	Implementation Class UID	UI	18	1.2.804.114118.3
0002 0013	Implementation Version Name	SH	6	
0002 0016	Source Application Entity Title	AE	18	
0002 0100	Private Information Creator UID	UI	0	(empty)
0002 0102	Private Information	OB	0	(binary data)
0008 0005	Specific Character Set	CS	12	ISO_IR 100
0008 0008	Image Type	CS	24	ORIGINAL\PRIMARY\OTHER
0008 0012	Instance Creation Date	DA	10	20100419
0008 0013	Instance Creation Time	TM	14	140916.000000
0008 0016	SOP Class UID	UI	26	1.2.840.10008.5.1.4.1.1.7
0008 0018	SOP Instance UID	UI	54	1.2.826.0.1.3680043.2.1208.34948545565201041914916296
0008 0020	Study Date	DA	10	20100419
0008 0021	Series Date	DA	10	20100419
0008 0030	Study Time	TM	14	140916.000000
0008 0031	Series Time	TM	14	140916.000000
0008 0032	Acquisition Time	TM	14	140916.000000
0008 0033	Content Time	TM	14	140916.000000
0008 0050	Accession Number	SH	2	
0008 0060	Modality	CS	4	OT
0008 0064	Conversion Type	CS	4	WSD
0008 0070	Manufacturer	LO	4	IIS
0008 0080	Institution Name	LO	2	
0008 0090	Referring Physician's Name	PN	2	

Figura 1.2: Esempio di attributi DICOM

1.4 DICOM e il paradigma Object Oriented

Abbiamo già definito DICOM come uno standard di comunicazione d'immagini biomediche, prevalentemente radiologiche, complete delle relative informazioni, ma DICOM è in realtà più di un insieme di norme fissate per stabilire uno scambio di informazioni; lo standard realizza un esplicito e dettagliato modello di descrizione di una serie di oggetti (paziente, immagine,..) che formano il dato radiologico, e di come essi sono tra loro collegati. Alla base dei protocolli definiti da DICOM esiste un modello del mondo reale, cioè un modello di come le diverse attività ospedaliere (in particolare quelle radiologiche) si svolgono nell'ambiente operativo. L'approccio a sviluppare strutture di dati basate su modelli e analisi di versioni astratte di entità reali è la cosiddetta struttura orientata ad oggetti la quale offre il grande vantaggio di mostrare chiaramente, sia i dati richiesti, sia le modalità di interazione e correlazione tra di essi. Definiti gli oggetti di interesse e tutte le loro caratteristiche, DICOM definisce quali operazioni possono essere eseguite e su quali oggetti. Tali operazioni sono chiamate DIMSE service (Dicom Message Service). La combinazione di un oggetto ed i corrispondenti servizi prende il nome di SOP (Service Object Pair). L'insieme delle SOP relative ad un unico oggetto formano una SOP Class. -

1.5 Comunicazione nello standard DICOM

Chiarita l'unità funzionale fondamentale definita da DICOM (la SOP Class) vediamo come avviene lo scambio di informazioni. Alla base del protocollo in esame esiste un approccio Client/Server, nel senso che, ogni volta che due applicazioni decidono di connettersi per scambiarsi informazioni, una delle due deve svolgere il ruolo di fornitore del servizio (SCP: Service Class Provider) mentre l'altra quello di utente (SCU: Service Class User). Ovviamente, per ciascuna combinazione di SOP Class e ruolo, lo standard definisce un set di regole di base controllate in fase di pre-colloquio o negoziazione, durante la quale si stabilisce se la comunicazione tra i due apparati è possibile oppure no.

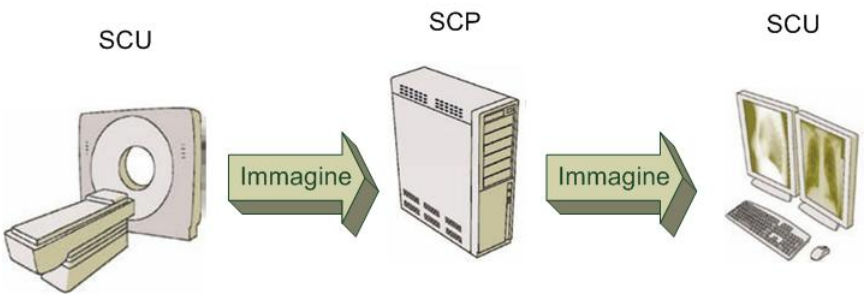

Figura 1.3: Esempio di comunicazione DICOM

1.6 Sicurezza nel DICOM

La sicurezza è un aspetto molto sentito nello standard DICOM, anche perché è uno dei principali motivi per i quali si spinge verso l'adozione di uno standard generale e uguale per tutti. Il DICOM parla di questo aspetto in ben 2 delle 18 parti del quale è composto e ne delinea i tratti fondamentali. La politica di DICOM di adottare standard affermati e riconosciuti si riflette nella scelta di ben noti algoritmi crittografici (RSA, SHA-1, AES..) e protocolli di comunicazione (TLS3, ISCL4).

Allo stato attuale la Sicurezza DICOM è assicurata da:

- Uso di comunicazioni sicure
  - Integrità dei dati durante la trasmissione
  - Autenticazione delle entità coinvolte
  - Confidenzialità durante la trasmissione attraverso l'encryption
- Sicurezza DICOM attraverso Buste Crittografiche CMS
  - Verifica dell'integrità dei dati
  - Confidentialità attraverso l'encryption
  - Accesso autorizzato
- Firme Digitali
  - Integrità dei Dati per l'intera vita di una istanza SOP
  - Identificazione di chi firma, con timestamp opzionali

1.6.1 Firme Digitali

Vediamo ora come il DICOM protegge i suoi file da manomissioni attravarso l'uso delle firme digitali. Lo standard spende una delle sue parti su quesEo argomento su questo e spiega come deve essere una firma digitale, quali algoritmi crittografici utilizzare e dove archiviare il tutto.

Il primo passo da compiere per firmare digitalmente un file DICOM è quello di calcolarne l'hash con uno dei seguenti algoritmi, esplicitamente indicati dallo standard:

- RIPEMD-160;
- MD5;
- SHA-1;

Tutti e tre gli algoritmi prendono come input il file DICOM e danno come output una stringa di bit di lunghezza dipendente dall'algoritmo utilizzato (ad esempio RIPEMD-160 produce una stringa di 160 bit) dimodoché piccolissime variazioni nel file producano stringhe completamente diverse. In effetti, è stato dimostrato sperimentalmente che la probabilità che due file diversi diano come risultato la stessa stringa di bit è assolutamente insignificante.

A questo punto bisogna criptare l'hash ottenuto nel precedente passo con un algoritmo di cifratura a chiave pubblica (o assimettrica). Il DICOM consiglia fortemente l'uso dell'algoritmo RSA (Rivest Shamir Adleman) il quale necessita di una coppia di chiavi, una per criptare l'hash (chiave privata) e una per decriptarlo (chiave pubblica). La chiave pubblica associata alla chiave privata RSA utilizzata per firmare deve essere trasmessa attraverso un certificato X.509. Lo standard DICOM ha riservato degli attributi per contenere la firma digitale e altre informazioni inerenti al processo di firma come l'algoritmo utilizzato per calcolare l'hash del file o il tipo di certificato X.509 trasmesso. Chiunque può estrarre la firma digitale dall'apposito attributo, decriptare l'hash del file con la chiave pubblica, ricalcolare l'hash del file e verificarne l'integrità confrontandolo con l'hash pervenuto assieme allo stesso.

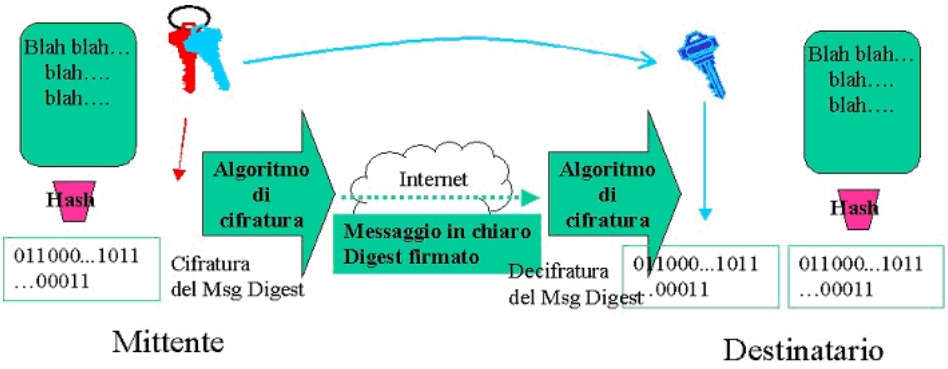

Figura 1.4: Firma Digitale

1.7 Software DICOM

La diffusione dello standard DICOM ha contribuito alla proliferazione di software DICOM che consentono di visualizzare o manipolare le immagini diagnostiche. Oltre ai software proprietari dei produttori di dispositivi DICOM è facile reperire software DICOM sviluppato da produttori software commerciali o anche nell'ambito di progetti Open source, che permettono di visualizzare e manipolare immagini DICOM senza alcun costo aggiuntivo.

1.8 Vantaggi dello standard DICOM

Lo standard DICOM ha avuto un grande successo, divenendo di fatto il riferimento per l'acquisizione, la comunicazione e l'archiviazione delle radiografie digitali. Ecco alcuni dei principali motivi:

- è uno standard internazionale e di conseguenza si riducono le problematiche inerenti al trasferimento di file;
- rende il processo di acquisizione e archiviazione indipendente dalla piattaforma operativa anche grazie all'enorme disponibilità di software DICOM.
- rende indivisibili l'immagine e dati riguardanti l'immagine (paziente, medico curante, studio, ...) senza alcun rischio di perdere o scambiare informazioni;
- rende possibile l'aggiunta d'informazioni senza limiti di spazio;
- utilizza lo standard TCP/IP per la comunicazione;
- utilizza i più comuni algoritmi per la creazione di firme digitali sicure e a prova di contraffazione.

Questi sono solo alcuni dei vantaggi che porta l'acquisizione di uno standard come il DICOM e possiamo dunque concludere che in futuro sarà sempre più utilizzato fino a diventare un giorno (forse) l'unico mezzo per acquisiSire, archiviare e trasmettere radiografie digitali.

Capitolo 2

Watermark

2.1 Introduzione

Il termine inglese watermarking si riferisce all'inclusione di informazioni all'interno di un file multimediale o di altro genere, che può essere successivamente rilevato o estratto per trarre informazioni sulla sua origine e provenienza. Per mezzo del watermark il documento è ancora accessibile, ma contrassegnato in modo permanente.

Tali indicazioni, dette watermark, possono essere evidenti per l'utente del file (per esempio nel caso di una indicazione di copyright applicata in sovraimpressione su una immagine digitale) o latenti (nascoste all'interno del file); in quest'ultimo caso il watermarking può essere considerato una forma di steganografia. La tecnica del watermarking digitale può venire utilizzata con diversi scopi:

- rendere manifesto a tutti gli utenti chi sia il legittimo proprietario del documento (nel caso in cui il marchio sia visibile);
- dimostrare l'originalità di un documento non contraffatto;
- evitare la distribuzione di copie non autorizzate;
- marcare alcune caratteristiche specifiche del documento;
- segnare il percorso di vendita del documento, utilizzando un marchio differente per ciascun acquirente.

2.2 Classificazione dei Watermark

I watermark possono essere classificati a seconda di alcune loro proprietà, che dipendono dallo scopo con cui sono stati inseriti all'interno del documento.

2.2.1 Visibilità

Una delle caratteristiche che differenzia i watermark è la visibilità; infatti possono essere **visibili** o **invisibili** a seconda delle necessità dell'autore. Nel primo caso viene utilizzato per dare un certo tipo d'informazione all'utente finale, che può quindi ricavare quest'ultima semplicemente guardando il documento digitale, senza alcun tipo di operazione aggiuntiva. Il watermark visibile viene spesso utilizzato per chi vuole marchiare un proprio lavoro (video, immagine, ...) con un segno di riconoscimento, dimodochè il copyright sia garantito e il documento resti eternamente associato ad un individuo o società.

Il watermark **invisibile** è utilizzato invece per nascondere informazione all'utente finale e spesso è preferito all'altro perchè la copia marcata è quasi identica all'originale, a meno di alcune differenze non riscontrabili dalle percezioni umane. Per ricavare l'informazione nascosta nel documento digitale è necessario effettuare operazioni aggiuntive che dipendono dall'algoritmo utilizzato. Questo tipo di watermark è impiegato per aggiungere informazioni all'immagine come ad esempio l'identità dell'autore, percorsi di vendita, firma digitale del documento, ...

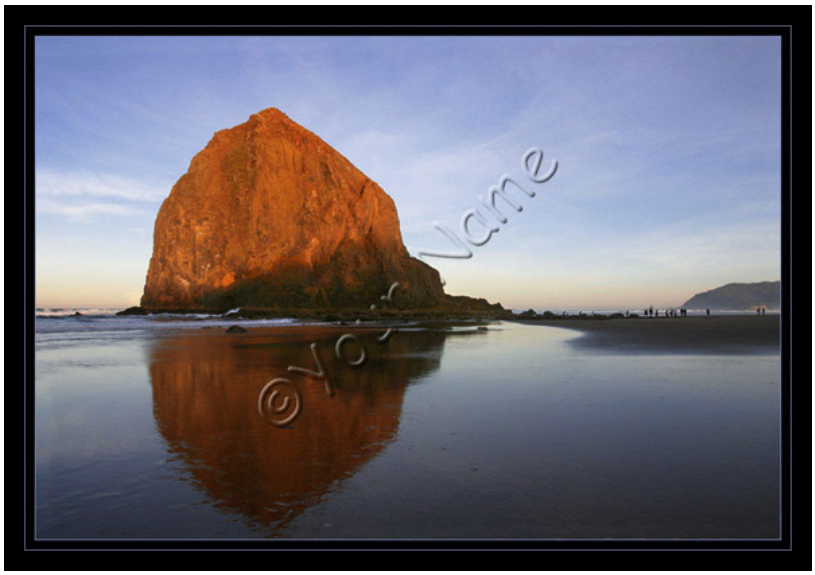

Figura 2.1: Esempio di Watermark visibile

2.2.2 Capacità di Resistenza

I watermark possono essere caratterizzati dalla loro capacità di resistenza agli attacchi: un watermark **fragile** può essere facilmente attaccato, distrutto e reso irriconoscibile da quasi ogni tipo di manipolazione dei dati. Esso è concepito per quelle applicazioni in cui si desidera sapere se una certa informazione è stata modificata nel passaggio dal creatore all'utilizzatore; nel qual caso il watermark non deve essere rilevabile o, comunque, deve presentare alterazioni.

Un watermark **semifragile** subisce la stessa sorte di quello fragile se i cambiamenti inflitti sono superiori a una certa soglia definita dall'utente. Questo tipo di watermark è spesso utilizzato nelle applicazioni in cui si vuole rivelare un certo tipo di manipolazioni.

Infine un watermark **robusto** deve resistere alle più comuni operazioni e trasformazioni sui dati, in quanto è utilizzato quando la proprietà del documento deve essere provata o garantita. L'informazione che trasporta non deve perdersi e deve potersi recuperare, anche se il documento è stato modificato. Inoltre si tende a considerare un watermark robusto quando è in grado di resistere anche ad attacchi intenzionali volti alla sua rimozione.

2.2.3 Necessità dell'originale

I watermark possono essere **ciechi** se per verificare la loro presenza non è necessario il documento originale; sono **semi-ciechi** se per essere rilevati hanno bisogno del watermark originale e sono **non ciechi** quando hanno bisogno del documento originale. Il vantaggio principale dei watermark non ciechi è il fatto che sono molto più robusti, ma spesso non è possibile avere l'originale per la verifica per cui sono poco utilizzati.

2.2.4 Reversibilità

Un watermark si definisce **reversibile** se dall'immagine marcata si riesce a tornare all'immagine originale e quindi di fatto, conoscendo l'algoritmo, si riesce a togliere il watermark. Al contrario un watermark si definisce **irreversibile** se non si riesce a tornare all'immagine originale da quella marcata. Solitamente, il primo tipo di watermark viene utilizzato per applicazioni che necessitano di verificare l'integrità del documento stesso. Il secondo tipo invece viene impiegato quando è necessario che un watermark resti sempre all'interno del documento (ad esempio quando si vuole preservare il copyright).

2.3 Tecniche di inserimento del Watermark nelle immagini

Esistono diverse tecniche per inserire un Watermark all'interno di un'immagine e sono principalmente raggruppabili in due gruppi dei quali ora vedremo le principali caratteristiche.

2.3.1 Tecniche basate sul dominio spaziale

Le tecniche basate sul dominio spaziale modificano direttamente i valori dei pixel dell'immagine, in base al codice che deve essere incluso. Generalmente considerano l'immagine come una matrice formata da **N X M** pixel e su questa effettuano le operazioni per inserire il watermark. Vale la pena rimarcare che i watermark inseriti con questo tipo di tecnica non sono particolarmente robusti, poichè agiscOno direttamente sui valori dei pixel. Di conseguenza sono spesso utilizzati per l'inserimento di watermark fragili, ai fini di rivelare manomissioni al documento. Le più comuni tecniche utilizzano una di queste 2 metodologie:

Regioni di non interesse

Questo tipo di metodologia nasconde il watermark nelle regioni di non interesse dell'immagine (RONI, Region Of Non Interest), in modo tale da non compromettere l'immagine stessa. Vari esperimenti nel campo della radiografia digitale hanno dimostrato che solitamente queste RONI corrispondono allo sfondo nero dell'immagine. Visto il fatto che il watermark è nascosto in regioni che non hanno alcun tipo d'interesse, il requisito d'invisibilità (per i watermark che devono essere invisibili) viene meno e di conseguenza aumenta lo spazio disponibile per le informazioni da nascondere. Un grave difetto di questo approccio è che non sempre si riescono ad individuare all'interno di un'immagine delle regioni di non interesse e, in qualsiasi caso, spetta all'operatore definirle rendendo questa tecnica troppo soggettiva.

Figura 2.2: Esempio di Regione di non Interesse

Least Significant Bit

Questo tipo di metodologia, al contrario della precedente, non fa distinzioni di regione ma cerca di nascondere il watermark nell'ultimo bit di ogni pixel (ad esempio, se un'immagine è codificata con 8 bit per pixel, questa tecnica usa l'ottavo bit). In tal modo la distorsione introdotta è minima, dimodochè l'occhio umano non possa percepire la differenza tra immagine originale e immagine marcata. Il vantaggio rispetto alla tecnica RONI consiste nel fatto che il meccanismo d'inserimento del watermark non necessita dell'intervento umano. Inoltre, questa tecnica non richiede l'uso di partIcolari regioni, indi per cui è attuabile su qualsiasi immagine. Ha come controindicazione il fatto che introduce distorsione (seppur minima) in parti importanti dell'immagine e, ad esempio, in campo medico può risultare un grosso problema. In letteratura sono state proposte diverse tecniche che utilizzano il Least Significant Bit, sia reversibili che irreversibili ma lo studio di esse esula dalle intenzioni di questa relazione.

2.3.2 Tecniche basate sulle trasformate

Nelle tecniche basate sulle trasformate si utilizzano alcune delle più comuni trasformate sui pixel e il watermark viene incorporato nei coefficienti della trasformazione. Alcune delle trasformate utilizzate sono:

- la **trasformata discreta del coseno**;
- la **trasformata discreta di Wavelet**: Permette di suddividere il segnale discreto in quattro sottobande di frequenza diverse. La percezione umana riesce a carpire le informazioni dalla sottobanda con le frequenze più basse, perciò il watermark è nascosto nelle altre tre sottobande in modo tale da non compromettere il documento. É la più utilizzata nel campo del watermark (Figura 2.3);
- la **trasformata discreta di Fourier**.

Tali tecniche operano seguendo questo schema: l'immagine viene convertita in una matrice in cui sono riportati i valori numerici (luminosità) dei singoli pixel; a questa matrice è applicata una delle trasformazioni invertibili sopra citate. Alcuni dei coefficienti della trasformata vengono modificati, ottenendo l'inserimento del watermark. Infine, applicando la trasformata inversa, si ricompone una matrice, e quindi un'immagine, simile a quella di partenza, a cui, però è stato applicato il marchio.

I watermark inseriti con questo tipo di tecnica sono particolarmente robusti grazie alle proprietà delle trasformate.

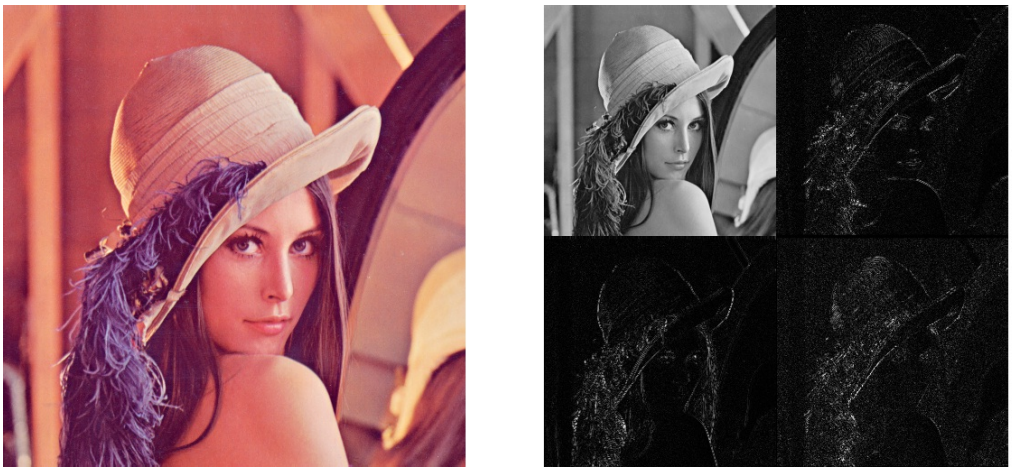

Figura 2.3: Alla sinistra l'immagine originale e alla destra l'immagine alla quale è stata applicata la trasformata discreta di wavevelet.

2.4 Watermark nelle immagini

Consideriamo, ora, solo immagini fisse, quali le fotografie e i disegni digitali.

Un qualunque schema di watermarking è realizzato attraverso l'implementazione di due ben specifici algoritmi: uno di codifica del marchio, che prende in input immagine originale e ne restituisce in output la corrispondente immagine opportunamente marcata e il marchio vero e proprio; l'altro di decodifica che, presa in input l'immagine marchiata (e l'immagine originale se il watermark è non cieco), restituisce il marchio associato. L'aggiunta di un watermark all'immagine può essere vista come l'inserimento di una componente di rumore nell'immagine stessa.

Chiamiamo V l'immagine originale, $W = \{w_1, w_2, \ldots, w_n\}$ il watermark da inserire (che potrebbe dipendere da V) e V_w l'immagine marchiata. V_w si può ottenere da V e W tramite un'opportuna funzione di codifica E:

$$E(V, W) = V_w$$

La funzione di decodifica D (nel caso di watermark cieco) vuole in ingresso un'immagine marchiata Vw e restituisce il watermark W':

$$D(V_w) = W'$$

Mentre se il watermark è non cieco lo schema va modificato nel modo seguente:

$$D(V, V_w) = W'$$

Nel caso di watermark robusti e quindi progettati per resistere agli attacchi W e W' non sono necessariamente identici, in quanto l'immagine può essere stata modificata tra la fase di codifica e di decodifica. È quindi necessaria una funzione di comparazione C_δ che permetta di stabilire se i due watermark corrispondono:

$$C_\delta(W, W') = c$$

- se $c > \delta$, W e W' corrispondono,
- altrimenti, W e W' non corrispondono

dove δ è un valore di soglia opportunamente stabilito.
Nel caso di watermark fragili e quindi progettati per stabilire se un'immagine è stata modificata o meno W e W' devono essere uguali.
In generale per codificare un watermark si scelgono alcune particolari caratteristiche dell'immagine, dette $F = \{f_1, f_2, \ldots, f_n\}$, a cui si applica un operatore di inserimento $\oplus$:

$$f'_i = f_i \oplus w_i \quad con \quad i = 1, \ldots, n$$

dove $F' = \{f'_1, f'_2, \ldots, f'_n\}$ sono le caratteristiche dell'immagine con watermark.
Il watermark viene decodificato con un operatore di estrazione $\ominus$, inverso rispetto al precedente, tale che:

$$w'_i = f'_i \ominus f_i \quad con \quad i = 1, \ldots, n$$

dove $W' = \{w'_1, w'_2, \ldots, w'_n\}$ è il watermark estratto.

Nel caso delle tecniche sul dominio spaziale le caratteristiche F sono i valori dei pixel dell'immagine; mentre nel caso delle tecniche sulle trasformate sono i valori dei coefficienti di una trasformata di dominio dell'immagine. L'insieme F è scelto in modo che piccole modifiche su ogni caratteristica non peggiorino sensibilmente l'immagine (il watermark può essere inserito senza un visibile danneggiamento dell'immagine) e che ogni caratteristica non cambi significativamente a meno che l'immagine non sia stata modificata in modo percettibile (il watermark deve poter essere decodificato senza ambiguità).

2.5 PSNR

Il peak signal-to-noise ratio (spesso abbreviata con PSNR) è una misura adottata per valutare la differenza tra due immagini. É utilizzata nel campo del watermark per verificare la bontà di una tecnica d'inserimento: tanto più è alto questo valore tanto più l'immagine sarà simile all'originale. Questo indice di qualità delle immagini è definito come il rapporto tra la massima potenza di un segnale

e la potenza di rumore che può invalidare la fedeltà della sua rappresentanzione. Poiché molti segnali hanno una gamma dinamica molto ampia, il PSNR è solitamente espresso in termini di scala logaritmica di decibel. È più facile da definire attraverso l'errore quadratico medio (MSE). Denotando con I l'immagine originale e con K l'immagine marcata, entrambe di dimensione MxN si definisce MSE (Mean Square Error) tra le due immagini:

$$MSE = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \|I(i,j) - K(i,j)\|^2$$

Il PSNR è definito come:

$$PSNR = 20 \cdot \log_{10} \left(\frac{MAX\{I\}}{\sqrt{MSE}} \right)$$

Dove $MAX\{I\}$ è il massimo valore del pixel dell'immagine. Per una immagine binaria il numeratore vale 1, per una immagine a livelli di grigio il numeratore vale 255. Per immagini a colori tale definizione vale per una componente. Vale la pena sottolineare il PSNR non è una misura assoluta ma serve solo a confrontare diverse tecniche di watermark. In generale, tecniche con valori di PSNR superiori ai 30 db sono considerate ottimali.

2.6 Watermark e DICOM

In letteratura sono stati proposti diversi algoritmi per inserire watermark all'interno di file DICOM sia nel dominio spaziale che in quello delle trasformate. In questo campo l'inserimento di watermark è questione ben più delicata rispetto ad altre situazioni, perchè qualora un medico debba fare una diagnosi sull'immagine marcata essa deve essere identica all'originale per quanto concerne la percezione umana. Di conseguenza il watermark deve essere assolutamente invisibile e quindi avere un alto PSNR. Come già anticipato molti sono stati gli algoritmi proposti ma tutti devono avere delle caratteristiche comuni che andremo ad elencare:

- l'invisibilità: i watermark devono essere assolutamente invisibili per non rischiare di inficiare la diagnosi del medico;
- fragilità: in questo campo ciò che conta è poter comprovare la non manomissione e l'autenticità delle immagini. Di conseguenza è doveroso utilizzare watermark di tipo fragile, dimodochè una qualsiasi modifica (intenzionale o non) venga scoperta;
- cecità: quasi sempre non è possibile avere l'immagine originale non marcata, perciò è necessario l'uso di algoritmi (ciechi) che non la richiedano per l'estrazione e la successiva verifica del watermark.

- reversibilità: sarebbe ottimale che dall'immagine marcata si riuscisse a togliere il watermark in modo tale da tornare all'immaginbe originale. Questo requisito è opzionale ma certamente desiderabile, soprattutto per il campo medico.

Capitolo 3

DicomPlayMaker

3.1 Introduzione

Alla NSB (Network Solutions for Business) di Altavilla Vicentina (Vi) ho collaborato alla progettazione e creazione di una macchina proprietaria (Appliance), la quale fosse in grado di:

- dialogare con i dispositivi radiografici per l'acquisizione delle immagini,
- rielaborare queste ultime per produrre un file DICOM con i relativi attributi,
- apporre un marchio per la comprovazione dell'autenticità,
- archiviare i file DICOM.

Questo obiettivo è stato pienamente raggiunto con un macchina basata su kernel Microsoft (del quale sono state tolte tutte le interfacce grafiche) e installata su mini-PC ASUS. Grazie ai driver forniti dai produttori dei radiografici è stato possibile dialogare con essi ed acquisire le immagini. Sono state poi utilizzate delle libreria open-source per l'elaborazione delle immagini e la costruzione dei file DICOM. Infine, utilizzando le indicazioni dello standard DICOM per l'archiviazione dei file è stato possibile costruire un database all'interno della macchina stessa.

Figura 3.1: Funzionamento di DicomPlayMaker

3.2 Sicurezza in DicomPlayMaker

Sono stato incaricato di sviluppare la parte sulla sicurezza di DicomPlayMaker. Sin dalle valutazioni iniziali era emersa la necessità di fornire all'utente finale una metodologia che gli permettesse di comprovare l'autenticità e, possibilmente, la non ripudiabilità del file DICOM. Dopo aver studiato a fondo la parte sulla sicurezza dello standard DICOM e aver appreso la metodica con la quale lo standard consiglia di rendere sicuri i file DICOM, ci siamo resi conto che era insufficiente per le nostre necessità. Questo perchè lo standard consiglia di aggiungere la firma digitale del file nell'header dello stesso, negli attributi che sono stati riservati allo scopo. Ciò comporta però un problema piuttosto serio: chiunque può togliere la firma digitale semplicemente eliminando il relativo attributo e questo può essere fatto in maniera non intenzionale. Serviva perciò qualcosa di più sicuro, che ravvisasse senza ambiguità una manomissione intenzionale dell'immagine. Ecco dunque che siamo giunti alla soluzione proposta da alcuni ricercatori, ovvero includere la firma digitale nell'immagine stessa con la tecnica del watermark.

3.3 Lìalgoritmo proposto da Zaini e Clarke

Restava ora da capire quale tecnica utilizzare e quale algoritmo implementare. Lavorando in campo medico, il watermark doveva categoricamente soddisfare i 4 punti precedentemenete elencati: invisibilità, fragilità, cecità e se possibile, reversibilità. Sin da subito, perciò, è stato possibile escludere tutte le tecniche che operavano nel dominio delle trasformate, a causa del fatto che i watermark inseriti con queste metodologie sono robusti. Sono state vagliate anche molte tecniche operanti nel dominio spaziale ma la maggior parte introduceva nell'immagine una distorsione inacettabile per il campo medico. Sono giunto dunque a dover decidere tra 2 tecniche proposte da due gruppi differenti di ricercatori, una che utilizza le regioni di non interesse e l'altra il Least Significant Bit.

La prima tecnica, presentata da Jasni Mohamad Zaini (ricercatore malese) e Malcolm Clarke (ricercatore inglese), propone di individuare un'area di non interesse, calcolare la firma digitale dell'immagine e inserire i bit che la compongono nell'area precedentemente individuata. Con questo schema operativo è necessario comunicare al destinatario la posizione dei bit della firma digitale, introducendo una prima difficoltà: quest'informazione aggiuntiva non è facilmente trasmissibile, a meno di individuare una zona dell'immagine atta a contenere questi dati. Inoltre, come già precedentemente sottolineato, l'area nella quale inserire i bit della firma digitale deve essere tracciata dall'operatore, rendendo l'algoritmo non completamente automatizzato. In ragione di queste due problematiche l'algoritmo è stato scartato.

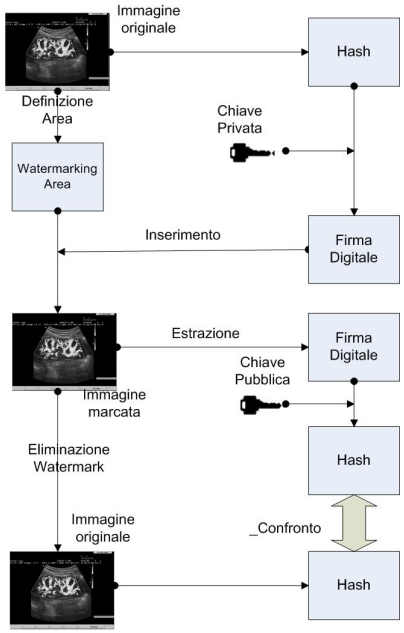

Figura 3.2: L'algoritmo proposto da Zaini e Clarke

3.4 L'algoritmo proposto da van Leest

L'algoritmo che mi è sembrato più adatto alle nostre esigenze è quello proposto da un gruppo di ricercatori olandesi dell'università di Eindoven, guidati dal professor Arno Van Leest. É stato pubblicato nel 2004 in un articolo intitolato Reversible Watermarking for Images. Quest'algoritmo opera nel dominio spaziale e usa la tecnica del Least Significant Bit. Il vantaggio più ecclatante è il fatto che introduce pochissima distorsione nelle immagini, assolutamente invisibile all'occhio umano. Inoltre rispetta anche le altre 3 caratteristiche che un watermark deve avere nel campo della radiografia digitale. In primis produce un tipo di watermark fragile, capace di individuare anche la modifica di un solo bit (grazie alle caratteristiche delle funzioni hash). In secundis per essere recuperato, il watermark inserito con questa tecnica non ha bisogno dell'immagine originale e tantomeno del watermark stesso rendendolo così cieco. Infine è un algoritmo reversibile, ovvero si può tornare in qualsiasi momento all'immagine originale da quella marcata. Dal nostro punto di vista, quest'algoritmo rappresenta la migliore soluzione per le nostre esigenze, per cui, dopo molti test, ho deciso di implementarlo nella DicomPlatMaker. É stato necessario apportare alcune modifiche all'algoritmo originale a causa di alcune mancanze nell'esposizione dello stesso che lo rendevano non implementabile nella pratica. Di seguito funzionamento, caratteristiche e conclusioni sull'algoritmo adottato.

3.4.1 Tecnica di compressione

Prima di iniziare a spiegare l'algoritmo è necessario esporre alcuni concetti della tecnica di compressione sulla quale è basato l'algoritmo in questione.
Supponiamo di avere un segnale tempo-discreto

$$x \in \{0, 1, \ldots, 2^m - 1\}^N = N_m^N$$

dove $m \in \mathbb{N}$ indica il numero di bit usati per ogni valore. Il segnale x è passato attraverso una funzione di compressione C_Q nella seguente maniera:

$$x_Q[n] = (C_Q x)[n]$$

dove x_Q rappresenta il segnale compresso. L'obiettivo di questa funzione è quello di mappare diversi valori di input $x[n]$ in uguali valori di output $x_Q[n]$. Come esempio di funzione C_Q possiamo considerare una semplice operazione di bit-shift (l'1 viene mappato in 2, il 2 in 4, il 3 in 6, ...) che da le basi per una buona funzione di compressione per un segnale audio. L'idea è di considerare i valori non utilizzati dalla funzione di mappatura (ad esempio i valori dispari dell'operazione di bit-shift) e in essi inserire una certa quantità d'informazione. Visto che il

sistema deve essere reversibile, bisogna trovare una funzione di espansione E_Q tale per cui:

$$(E_Q x_Q)[n] = x[n]$$

In generale questo però non è vero, perchè rimane sempre un errore q tale che:

$$q[n] = x[n] - (E_Q C_Q x)[n]$$

Per rendere dunque l'algoritmo totalmente reversibile è necessario inserire assieme al watermark l'errore q. Al decoder il segnale marcato è processato, in modo tale da estrarre il watermark e l'errore. La funzione E_Q è utilizzata per la mappatura inversa:

$$x_E[n] = (E_Q x_Q)[n]$$

Con una semplice aggiunta dell'errore q al segnale così estratto abbiamo un recupero bit per bit del segnale originale:

$$x[n] = x_E[n] + q[n]$$

3.4.2 Applicazione nelle immagini digitali

Nell'algoritmo in questione la funzione C_Q determina la quantità di distorsione dovuta all'inserimento del watermerk e lo spazio disponibile. Nella prossima sezione discuteremo come scegliere una buona funzione di compressione C_Q. Sottolineo inoltre, che tratteremo solo immagini a scala di grigi, ragion per cui quando parleremo di valore dei singoli pixel ci riferiremo alla loro luminosità.

Scelta della funzione C_Q

Per spiegare l'uso della funzione di compressione nelle immagine digitali, è utile vederle attraverso una prospettiva ad istogramma. Assumiamo che ogni immagine abbia un suo istogramma, rappresentante il numero di occorrenze per ogni valore (numero di occorrenze per il valore di luminosità 0, valore di luminosità 1, ...). Nell'esempio precedente, la funzione di bit-shift moltplica tutti i valori per due. Ciò significa che nel corrispondente istogramma i numeri dispari assumono valore zero. L'approccio bit-ahift usato, ottimo per un segnale audio, non è l'ideale per un'immagine digitale. La distribuzione di valori e le caratteristiche di un'immagine introdurrebbe forti distorsioni all'immagine originale, rendendola inutilizzabile.

Invece di creare gap in tutti i valori dispari, ne introdurremo solo alcuni in specifiche parti dell'istogramma. Il principio basilare è illustrato nella figura 3.3.

In questo esempio assumiamo un segnale intero x con l'istogramma corrispondente nella figura 3.3. Nella figura 3.4 è rappresentata una funzione di compressione C_Q che introduce un gap in corrispondenza del valore 6 che, non a caso, è vicino al valore di occorrenze massimo (il valore 5 ha 6 occorrenze). Siamo ora in grado di introdurre un segnale binario in questo modo:

- un bit di valore 1 viene inserito incrementando una delle occorrenze del valore 5 di una unità
- un bit di valore 0 viene inserito lasciando inalterata un'altra occorrenza del valore 5.

In questo esempio siamo in grado di inserire un totale di 6 bit cambiando tutte le occorrenze del valore 5 secondo l'ordine con il quale il segnale binario è codificato. In generale, è possibile inserire un numero di bit pari al numero massimo di occorrenze di un certo valore. Per il momento non tratteremo casi di overflow o underflow che saranno discussi successivamente. Il destinatario deve necessariamente sapere dove è stato introdotto il gap e ciò comporta un notevole problema. Infatti non è facile trasmettere quest'informazione e l'unica soluzione sembra essere quella di inserirla in una specifica posizione dell'immagine definita a priori e conosciuta da entrambe le parti. In questo modo il destinatario è in grado di trovare il valore di gap, recuperare il watermark è ricostruire l'immagine originale.

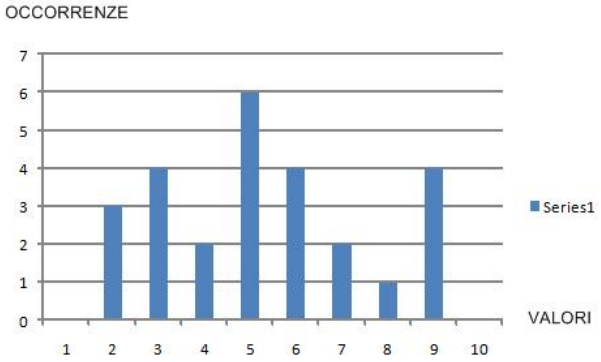

Figura 3.3: Istogramma immagine originale

Fuzione di Compressione Adattativa

Nell'esempio precedente abbiamo applicato la funzione di compressione in tutta l'immagine. Seguendo uno schema più sofisticato potremmo suddividere l'immagine in diversi blocchi con caratteristiche affini e applicare una funzione di compressione C_Q per ogni blocco. In effetti è stato provato sperimentalmente che questo algoritmo di inserimento del watermark è ottimizzato per piccoli blocchi

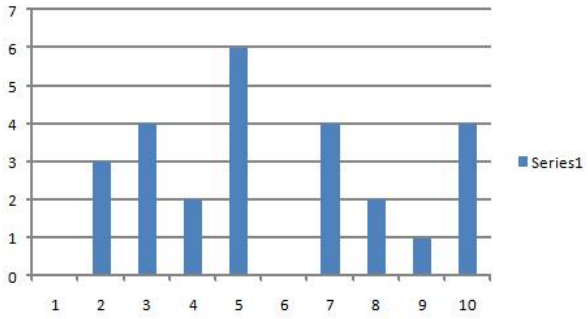

Figura 3.4: Istogramma immagine alla quale è stata applicata una funzione di compressione C_Q che ha introdutto un gap in corrispondenza del valore 5

dell'immagine. Un grosso svantaggio derivante dalla suddivisione dell'immagine è la necessità di introdurre informazione aggiuntiva, ovvero il valore di gap per ogni blocco.

In un'altra implementazione dell'algoritmo ho provato ad includere l'informazione aggiuntiva nel seguente modo: invece di creare il gap nel valore adiacente a quello con il numero di occorrenze massimo (nell'esempio precedente, il valore 6), ho introdotto il gap nel valore massimo e nel valore minino di luminosità presente nell'istogramma. Questo modo di procedere è raffigurato nelle figure 3.5 e 3.6. Il valore minimo, che in questo esempio è il 2 è decrementato di 1 e il valore massimo, che è rappresentato dal valore 9, è incrementato di 1. In tal modo è introdotto un gap nel valori massimi e minimi del segmento in questione. Seguendo questo approccio lo spazio disponibile per nascondere i bit del watermark dipende dal numero di occorrenze del valore massimo e del valore minimo. Resta però ancora un problema: può accadere infatti che il valore massimo e\o minimo cambino durante l'introduzione del watermark rendendo così impossibile per il destinatario sapere quale fosse il valore massimo e\o minimo effettivo. Per arginare questo problema bisogna che la prima occorrenza del valore massimo e\o minimo resti invariata, permettendo così al destinatario di conoscere i valori di gap. In tal modo si perde uno spazio per il valore massimo e uno spazio per il valore minimo. Per chiarire il tutto, nell'esempio precedente ci sono 3 occorrenze del valore minimo e 4 occorrenze del valore massimo; togliendo a queste le due occorrenze necessarie ad indicare i valori di gap abbiamo $3+4-2=5$ occorrenze disponibili, per un totale di 5 bit introducibili. É doveroso sottolineare come in questo esempio la distorsione sull'immagine originale sarà molto inferiore rispetto a quella dell'esempio precedente grazie al fatto che la maggior parte dei valori resta invariata.

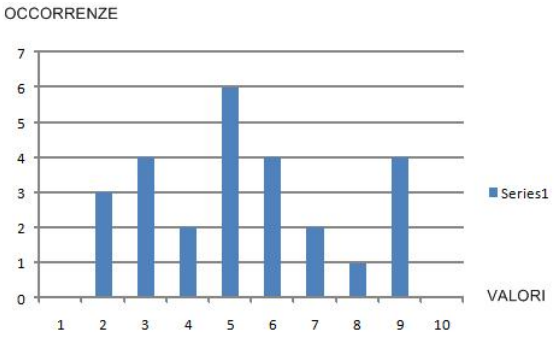

Figura 3.5: Istogramma immagine originale

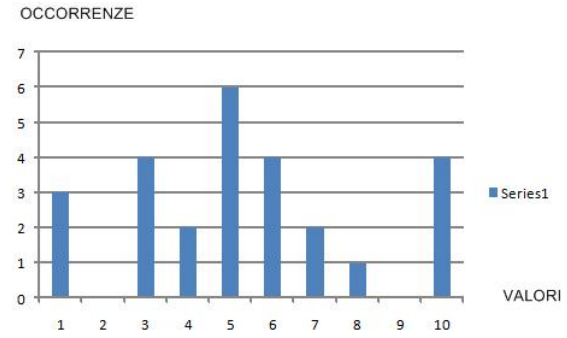

Figura 3.6: Istogramma immagine con il gap introdotto nel valore massimo e nel valore minimo

Esperimenti

É chiaro ormai come il secondo modo di procedere sia migliore rispetto al primo grazie al fatto che introduce minore distorsione. Inoltre, è sperimentalmente dimostrato che suddividere l'immagine in blocchi e applicare la funzione di compressione per ogni blocco aumenta lo spazio disponibile per l'inserimento dei bit formanti il watermark. Quindi non resta che scegliere la dimensione di questi blocchi. Ho testato varie dimensioni, a partire da blocchi formati da 16x16 pixel fino a quelli formati da 2x2 pixel. Senza dubbio i blocchi di quest'ultima dimensione sono quelli che hanno evidenziato un maggior spazio per l'inserimento del watermark. Anche in questo caso la scelta è stata piuttosto semplice. Ho notato inoltre come introducendo un solo valore di gap per ogni blocco (o il valore massimo o il valore minimo) migliori ulteriormente l'algoritmo grazie al fatto che con questa scelta la distorsione dell'immagine diminuisce ulteriormente. Introducendo un solo valore di gap otteniamo la seguente regola: per ogni blocco da 2x2 pixel è possibile introdurre al massimo 3 bit. Questo è speigato dal fatto che uno

serve a trasmetrere il valore di gap. Questo risultato si presenta quando tutti i pixel in un blocco hanno valore uguale. Al contrario, se i pixel di un blocco hanno valori tutti diversi tra loro non è possibile inserire alcun bit.

Underflow e Overflow

Non abbiamo ancora trattato il caso in cui il valore massimo (o minimo) assuma valore massimo (o minimo) in senso assoluto (0 e 255 per le immagini a 8 bit per pixel, 0 e 65535 per quelle a 16 bit per pixel). É chiaro che questi eventuali valori non possono essere utilizzati per creare un gap a causa del fatto che introdurrebbero un overflow (o underflow). Ciò significa che un pixel nero diventerebbe bianco e viceversa. Dobbiamo quindi escludere valori di questo tipo dalla selezione per i valori di gap. Questo fatto introduce un ulteriore problema: non si ha modo di sapere se valori massimi o minimi assoluti nell'immagine marcata fossero tali anche nell'immagine originale o al contrario siano frutto di un incremento (decremento) dovuto all'introduzione di un gap nel valore inferiore (o superiore) (1 e 254 per le immagini a 8 bpp, 1 e 65534 per quelle a 16 bpp). Ho risolto questo problema inserendo assieme al watermark la posizione di questi valori massimi (minimi) assoluti frutto dell'incremento (decremento). Il destinatario in tal modo può riportarli al loro valore originale una volta estratto il watermark.

3.5 Hash e Firma digitale

Il watermark in questione è formato dall'hash dell'immagine ottenuta grazie all'algoritmo RIPEMD-160, il quale produce una stringa di 160 bit. Una volta ottenuto l'hash, si procede con il criptaggio di quest'ultimo grazie all'algoritmo a chiavi simmetriche RSA. Quest'ultimo genera due chiavi, una privata e una pubblica tra loro associate univocamente. Con la chiave privata (consegnata al mittente) si cripta l'hash dell'immagine, ottenendo una stringa di 512, 1024 o 2048 bit (questo dipende dalla lunghezza delle chiavi). Questa stringa, che viene chiamata Firma Digitale, rappresenta il nostro watermark che verrà inserito nell'immagine. La chiave pubblica è invece trasmessa nell'header del file DICOM. Il destinatario estrae il watermark, lo decripta con la chiave pubblica ottenendo l'hash dell'immagine spedito dal mittente. Una volta fatto ciò riporta l'immagine marcata allo stato originale con la funzione inversa E_Q, calcola il nuovo hash e lo confronta con quello ottenuto in precedenza. Se i due coincidono allora l'immagine sarà autentica e non manomessa.

3.6 Riepilogo

Il mittente deve:

1. calcolare l'hash dell'immagine,
2. criptare l'hash con la chiave privata,
3. inserire il watermark:
   a) dividendo l'immagine in blocchi da 2x2 pixel,
   b) individuando il\i valore\i massimi per ogni blocco,
   c) incrementando di una unità i valori così trovati escludendo i blocchi nei quali quest'operazione potrebbe causare un overflow
   d) selezionando quei blocchi che hanno più di un'occorrenza del valore massimo (a causa del fatto che una serve sempre ad indicare il valore di gap).
   e) inserendo i bit nei blocchi (ove possibile) decrementando di 1 quando si deve inserire uno 0 e lasciando il valore invariato quando il bit è un 1*.
4. inserire le posizioni di eventuali valori massimi(minimi) assoluti frutto dell'incremento(decremento),
5. inserire la chiave pubblica nell'header del file DICOM.

Il destinatario deve:

1. estrarre il watermark:
   a) dividendo l'immagine in blocchi da 2x2 pixel
   b) individuando il valore massimo† per ogni blocco
   c) selezionando i blocchi che presentano una o più occorrenze di valori uguali o inferiori di 1 al valore massimo del blocco stesso,
   d) estraendo i bit in base a questa regola: se il blocco selezionato presenta una (o più) occorrenze con valore uguale al valore massimo sarà presente un (o più) 1, se al contrario il blocco presenta una (o più) occorrenze con valore uguale al valore massimo -1 sarà nascosto uno (o più) 0.
2. estrarre le eventuali posizioni di bit massimi(minimi) da incrementare(decrementare)
3. riportare l'immagine marcata all'originale con la funzione di espansione applicata ad ogni blocco,
4. calcolare l'hash dell'immagine così ottenuta,

*la prima occorrenza deve sempre restare invariata
†il valore massimo per ogni blocco s'intende il primo dei valori massimi

5. decriptare il watermark con la chiave pubblica presente nell'header ottenendo l'hash spedito,

6. confrontare l'hash così ottenuto con l'hash calcolato in precedenza.

3.7 Personalizzazioni e implementazione dell'algoritmo

L'algoritmo così presentato funziona molto bene ma dopo svariati test ho notato che potevano essere introdotte delle migliorie. Rispetto all'originale non ho utilizzato entrambi i valori (massimo e minimo) per la creazione del gap. In effetti creando solo un gap si guadagna notevolmente in termini di distorsione dell'immagine. Il secondo aspetto riguarda il fatto che il wateramark è formato da un numero di bit conosciuto a priori da entrambe le parti (mittente e destinatario), ragion per cui possiamo inserire i bit fino ad esaurimento e dall'ultimo blocco utilizzato in poi possiamo riportare l'immagine all'origine, limitando ancora di più la distorsione.

L'algoritmo è stato implementato nel linguaggio di programmazione C++. Sono state utilizzate diverse librerie open-source, tra le quali:

- le *DCMTK* 3.5.4 per la parte DICOM (estrazione dei pixel, salvataggio del file, ...);
- le *GDCM* per la decompressione delle immagini DICOM;
- le libreria standard della Microsoft per il calcolo dell'hash dell'immagine con l'algoritmo RIPEMD-160;
- le *CHILKAT* per la implementazione dell'algoritmo a chiavi simmetriche RSA.

Attualmente sono stati sviluppati separatamente tre eseguibili: uno per la parte d'inserimento del watermark, uno per la parte di verifica e uno per la generazione delle due chiavi private e pubbliche.

3.8 Test

Ho testato più di 150 file DICOM con questo algoritmo e l'esito è stato positivo per il 100% di essi. Ho utilizzato immagini a scala di grigi a 8 e 16 bit e immagini a colori RGB. Ho anche implementato l'algoritmo in modo tale che possa accettare tutti i più comuni formati di compressione delle immagini come *JPEG, RLE, JPEG2000, ...*

Il PSNR medio è stato di circa 46 db e confrontando ad occhio nudo non si nota alcuna differenza tra immagine marcata e immagine originale.

3.9 Conclusioni

L'implementazione dell'algoritmo è avvenuta con successo e il software che ne è risultsto sembra rispondere alle nostre esigenze iniziali. Le immagini non presentano distorsioni all'occhio umano, l'algoritmo è fragile e non ha la necessità dell'immagine originale per l'estrazione del watermark. Inoltre è completamente reversibile. Il calcolo del PSNR medio è più che soddisfacente.

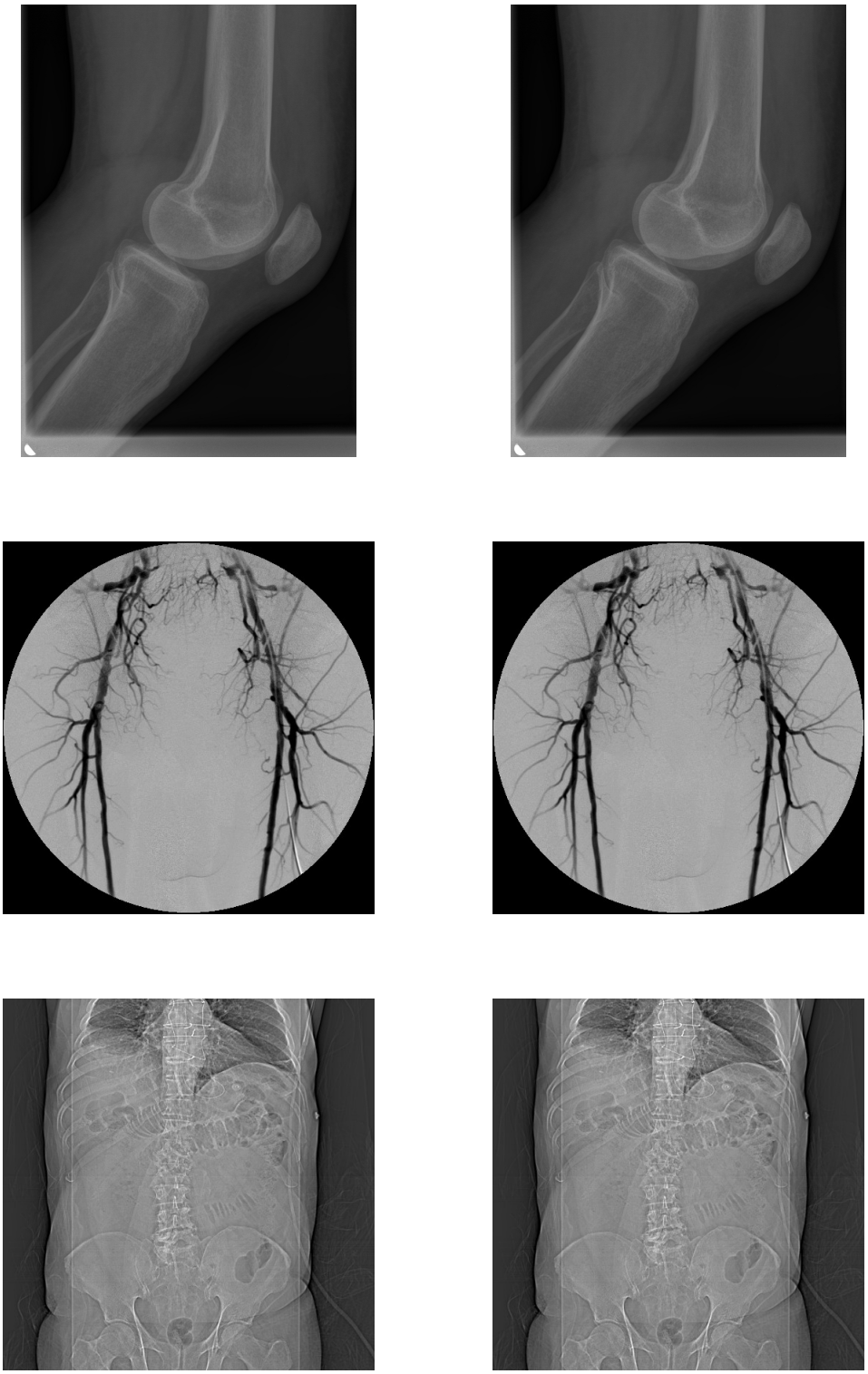

Figura 3.7: A sinistra le immagini originali e a destra quella marcate

Bibliografia

[1] F. B. A. van Leest, M. van der Veen. Reversible watermarking image. *Image Processing*.

[2] X. Guo and T. Zhuang. A region-based lossless watermarking scheme for enhancing security of medical data. *Journal of Digital Imaging*.

[3] O. Pianykh. *Digital Imagining and Communcations in Medicine (DICOM)*. Springer.

[4] G. C. W. Pan. Medical image integrity control combining digital signature and lossless watermarking. *Data Privacy Management and Autonomous Spontaneous Security*.

[5] J. Zain and M. Clarke. Reversible region of non-interest (roni) watermarking for authentication of dicom images. *International Journal of Computer Science and Network Security*.