

Grafica

Introduzione

Nella progettazione e nello sviluppo delle siti *Web* è ormai universalmente diffusa la presenza di elementi multimediali che hanno il duplice scopo di arricchire la pagine di nuovo contenuto informativo, veicolato attraverso *media* non testuali, e di renderle contemporaneamente più accattivanti.

L'aggettivo multimediale è in generale usato per indicare un sistema che si avvale di diversi tipi di *media*. Tipicamente i *media* utilizzati in una pagina *Web* sono il testo, le immagini, le animazioni (immagini in movimento), l'audio e il video. Tutti i tipi di *media* citati per essere utilizzati in ambito *Web*, devono avere un formato digitale. Questa prima caratteristica differenzia notevolmente i sistemi multimediali *Web based*, rispetto ai supporti multimediali disponibili prima della diffusione dell'ICT (*Information and Communication Technologies*) poiché è la base per la realizzazione di multimedia, che offrano, come il sistema che li ospita, un elevato grado di interattività.

Un segnale è in formato digitale se viene rappresentato sotto forma numerica, ovvero attraverso le cifre (in inglese *digit*) di un sistema di numerazione (decimale o binario). Per contrapposizione una grandezza è analogica se viene rappresentata mediante l'uso di altre grandezze legate alle prime da una relazione di analogia. Un tipico esempio d'uso di elementi digitali e analogici è l'orologio: l'orologio analogico, a lancette, rappresenta per analogia il tempo con lo spazio (il moto delle lancette). L'orologio digitale rappresenta invece il tempo mediante le cifre numeriche che visualizza sul *display*.

Tipologie di multimedia

I *media* digitali possono derivare da sorgenti reali di tipo analogico, come per esempio una foto passata allo *scanner*, o una immagine catturata da una fotocamera digitale. In questo caso il *media* digitale è frutto di un processo di digitalizzazione. Alternativamente esistono multimedia digitali di sintesi che non provengono da una fonte analogica ma sono stati creati direttamente al *computer*. Per esempio le animazione *Flash* così come tipicamente anche le icone, sono formati multimediali di sintesi.

E inoltre fondamentale distinguere tra:

- **Media discreti** (*time independent*): che sono identificabili con sequenze di elementi atomici o continui non dipendenti dal tempo. Sono *media* discreti testo e immagini.
- **Media continui** (*time dependent*): i cui valori cambiano nel tempo. La quantità di informazioni (*bit*) necessarie per rappresentare un *media* continuo cresce in funzione del tempo e per questo motivo i *media* continui sono gestiti in molti casi come flussi di informazioni, piuttosto che come *file*. Sono *media* continui audio, video e animazioni.

La realizzazione di elementi multimediali e interattivi può avvenire anche tramite l'esecuzione di *script* o programmi, come per esempio può accadere per le animazioni implementate come *applet Java*.

La trattazione relativa a audio e video è rinviata alla **seconda parte dell'Introduzione**. Nel seguito introdurremo invece i concetti base relativi a immagini e animazioni.

Quantizzazione e campionamento

La quantizzazione è un processo di approssimazione dei valori associati ai segnali analogici che vengono arrotondati per produrre i corrispondenti valori che codificano il segnale digitale. Infatti, i valori possibili per alcuni tipi di informazioni multimediali (ad esempio i colori di un'immagine o la frequenza di un suono) nel mondo reale sono infiniti (sono definiti nel continuo). Quando il *media* viene digitalizzato i valori disponibili sono in realtà una quantità finita (discreta) e dunque rispetto all'equivalente analogico, ogni segnale digitale è rappresentato da valori arrotondati.

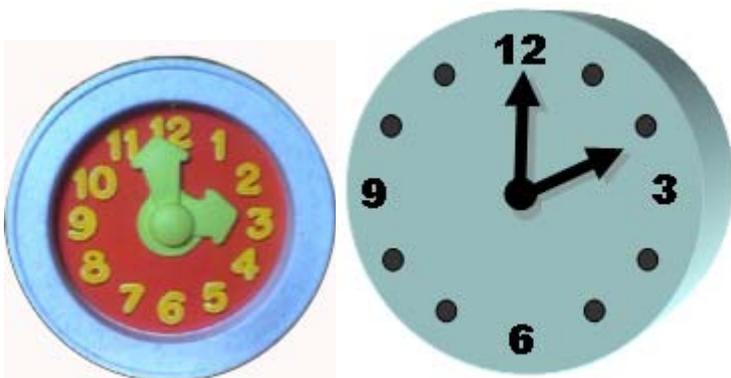
I valori possibili sono fissati in livelli di quantizzazione, ovvero un insieme di livelli essere tipicamente a intervalli regolari e in modo da non alterare in modo sensibile il segnale. Per esempio per le immagini vengono tipicamente catturati 256 colori (con 8 *bit*) oppure 2 colori (con 1 *bit*). Il numero di livelli di quantizzazione definisce il numero di *bit* necessari a codificare un campione del segnale.

Per digitalizzare un segnale continuo occorre anche misurare il livello di ogni campione ad intervalli discreti. Il numero di campioni in un intervallo di tempo è detto *sampling rate*, frequenza di campionamento e si misura in numero di volte al secondo ovvero in hertz. Esiste una frequenza di campionamento (detta valore di *Nyquist*) che garantisce la ricostruzione fedele del segnale ed è pari al doppio della frequenza massima del segnale stesso. Sottocampionare (ovvero campionare a frequenza minore del valore *Nyquist*) significa perdere informazioni e ottenere una ricostruzione del segnale imprecisa.

Immagini

Una immagine digitale può essere rappresentata per punti (immagine *bitmap* o *raster*) o in formato vettoriale. Le immagini *bitmap* sono costituite da una matrice di punti $n \times m$ ciascuno dei quali è associato ad un colore. Le immagini vettoriali sono invece costruite da descrizioni matematiche di uno o più elementi grafici, dette appunto vettori. Nella codifica delle immagini vettoriali vengono utilizzati non solo semplici vettori ma anche curve e archi.

Le figure che seguono mostrano due immagini che rappresentano un orologio, mediante una codifica per punti e mediante una codifica vettoriale. In realtà per essere inserita in questa pagina anche la versione vettoriale è stata convertita in formato *raster*.



I due formati presentano vantaggi e svantaggi: i vettori non possono essere usati facilmente per codificare immagini reali (come le fotografie) in cui il contenuto varia punto per punto e la resa (rappresentazione) può variare a seconda dall'applicazione usata per visualizzare l'immagine. D'altro canto nelle immagini vettoriali qualunque tipo di *zoom*, ridimensionamento o aumento di risoluzione, lascia inalterata la qualità dell'immagine, mentre nelle immagini *bitmap* queste operazioni ne degradano la qualità.

Bitmap: risoluzione spaziale

L'immagine *bitmap* è rappresentata da una matrice $n \times m$ di punti (*dot*) o *pixel* (quando si parla di monitor). Il numero di punti per pollice (*dot per inch*, dpi) è detto risoluzione e rapporta la dimensione dell'immagine nel reale (in *inch*) con la dimensione digitale (in *dot*). Il numero di dpi esprime una misura lineare che va rapportata alle due dimensioni indicando risoluzione verticale e orizzontale (di solito uguali).

Più punti per pollice si campionano, più la rappresentazione digitale sarà precisa e più grande (in *byte*) sarà il *file* risultante. Per esempio una immagine di 72 dpi utilizza $72 \times 72 = 5.184$ *pixel* per pollice quadrato. Se è di 3x3 pollici definisce $5184 \times 3 \times 3 = 46.656$ *pixel*. Dunque, se si usa:

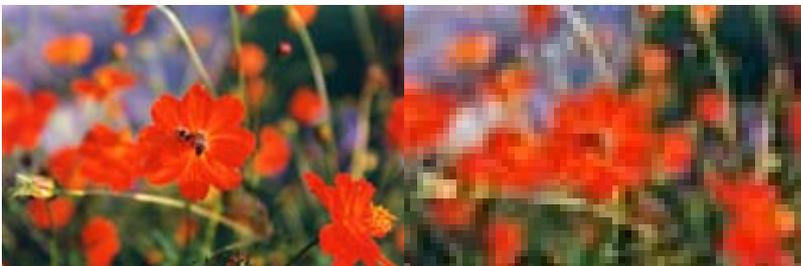
- 1 *bit* per ogni punto, occupa 5.832 *byte*.
- 1 *byte* per ogni punto, occupa 46.656 *byte*.
- 3 *byte* per ogni punto, occupa 139.968 *byte*.

Definizioni tipiche sono:

- Per lo schermo (bassa per la stampa) 72 dpi.
- Per la stampa (troppo alta per lo schermo) da 600 dpi in su.

Quindi per esempio una immagine che copra uno schermo *standard* di 800x600 punti (a 72 dpi) usando 3 *byte* per punto (16 milioni di colori) occupa 1.440.000 *byte*.

Le immagini che seguono mostrano la stessa immagine con risoluzione a 72 e a 36 dpi.



Bitmap: risoluzione cromatica

La codifica di ciascun punto dipende dal numero di valori scelti per quantizzarlo e sostanzialmente si utilizzano tre tipi di risoluzione cromatica: **Bicromia**, **Palette** e **True color**.

Bicromia: utilizza un *bit* per ogni punto per rappresentare solo bianco e nero. Possono essere utilizzate tecniche grafiche per ottenere rappresentazioni efficaci di immagini (ovviamente in B/N) anche in bicromia, accostando con opportuni algoritmi *dot* bianchi e *dot* neri. La figura che segue rappresenta una immagine d'esempio, memorizzata in bicromia.



La stessa immagine, memorizzata in bicromia ma prodotta con un opportuno algoritmo di *dithering*, ha una resa di questo tipo.



Palette: utilizza 8 *bit* per ogni punto (256 colori). I 256 colori possono essere scelti in molti modi: 256 toni di grigio per rendere l'immagine in B/N, 256 colori presi in modo uniforme sullo spettro oppure scegliendo con criteri diversi i 256 colori che offrono una migliore resa grafica dell'immagine. È in generale una risoluzione cromatica poco adatta alle immagini di tipo fotografico che sono caratterizzate da colori molto sfumati. Le immagini che seguono mostrano tre l'uso di tre *palette* sulla fotografia di esempio: *palette* di grigi (256 toni di grigio), *palette* di colori distribuiti uniformemente sullo spettro, *palette* con colori scelti in base ai colori contenuti nell'immagine.



True color: utilizza un sistema colorimetrico a tre o quattro valori e 8 *bit* per ogni valore (per un totale di 24 o 32 *bit* per ogni punto) che consente di rappresentare 16.777.216 colori. Il numero dei colori ottenibili supera quello dei colori effettivamente distinguibili dall'occhio umano, per il quale questa rappresentazione digitale è cromaticamente perfetta. È la risoluzione cromatica più adatta alla riproduzione di fotografie. La figura che segue mostra l'immagine d'esempio in *true color*.

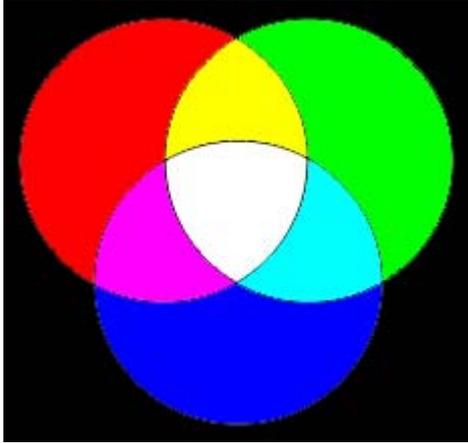


Spazi colorimetrici

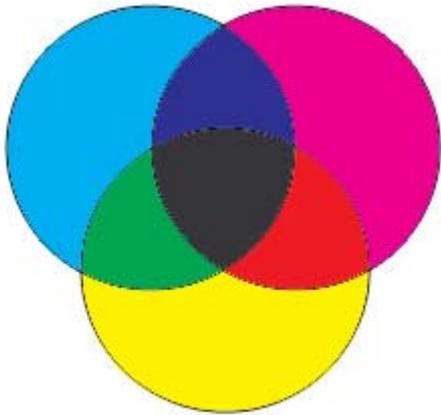
Nella rappresentazione digitale di immagini in *true color* si usa solitamente un sistema tricromatico:

- di tipo **additivo** (tipicamente monitor): i colori sono aggiunti al nero per creare nuovi colori; più colori sono aggiunti, più il colore risultante tende al bianco. La figura seguente mostra

uno spazio colorimetrico additivo.



- Di tipo **sottrattivo** (tipicamente stampanti): i colori sono sottratti dal bianco per creare nuovi colori; più colori sono tolti, più il colore risultante tende al nero. La figura seguente mostra uno spazio colorimetrico sottrattivo.



Alcuni spazi colorimetrici molto utilizzati sono i seguenti:

- **RGB Red-Green-Blue** (Rosso-Verde-Blu): è uno spazio colorimetrico additivo in cui i colori sono ottenuti componendo valori di intensità del colore di Rosso, Verde e Blu in un intervallo $[0,1]$. Il valore $(0,0,0)$ indica il nero mentre il valore $(1,1,1)$ indica il bianco.
- **CMY (Cyan-Magenta-Yellow)**: è uno spazio colorimetrico sottrattivo che si basa sull'assorbimento della luce da parte dell'inchiostro. Il valore $(0,0,0)$ indica il bianco mentre il valore $(1,1,1)$ indica il nero. Il nero composto nella stampa ha una cattiva resa, per cui a volte si utilizza una codifica **CMYK** con K che indica il nero puro (ottenuto da inchiostro nero).
- **YUV (Luminance-Chrominance)**: è una particolare codifica additiva che si ha per trasformazione lineare da quella **RGB** e che viene usata nelle trasmissioni televisive. Y contiene informazioni sulla Luminosità mentre U e V sono parametri che riguardano il colore.

Le seguenti equazioni riassumono i rapporti tra i colori definiti negli spazi colorimetrici **RGB** e **CMY**, indicati in questo modo R=*red* (rosso), G=*green* (verde), B=blu, C=ciano, M=magenta, Y=*yellow* (giallo), W=*white* (bianco):

- $C = G + B = W - R$
- $M = R + B = W - G$
- $Y = R + G = W - B$

Dimensioni e compressione

Abbiamo visto che la risoluzione spaziale incide con un fattore quadratico sulla dimensione in *byte* dell'immagine poiché il numero di *dot* della rappresentazione digitale viene considerato ovviamente nelle due dimensioni. Una stessa immagine di 3x4 *inch* richiede un numero di *dot* molto diverso se digitalizzata a 72 dpi o a 600 dpi:

- 72 dpi: 62.208 *dot*.
- 600 dpi: 4.320.000 *dot*.

A questa dimensione va aggiunta la risoluzione cromatica con un fattore moltiplicativo. La stessa immagine, con la stessa risoluzione spaziale può quindi richiedere occupazioni di memoria molto diverse. Supponiamo sia 72dpi e 3x4 *inch* (62.208 *dot*), servono:

- in bicromia: 7.776 *byte*.
- A palette: 62.208 *byte*.
- In True Color: 186.624 *byte*.

Esistono numerosi formati non compressi o lasciamente compressi, che vengono utilizzati per le applicazioni *stand-alone*, per la stampa o per conservare copie ad alta fedeltà di immagini che sono poi distribuite in forma compressa. Tra questi:

- **BMP (BitMaP)**: è il formato *standard* di *Microsoft Windows* e consente di usare diverse risoluzioni cromatiche (**RGB**, scala di colore e scala di grigio). Si può specificare una risoluzione in *bit* e per le immagini a 4 e 8 *bit* è possibile utilizzare un metodo di compressione *lossless*.
- **TIFF (Tagged Image File Format)**: è un formato *bitmap* supportato da quasi tutte le applicazioni grafiche, molto utilizzato perché consente di scambiare *file* tra programmi e piattaforme diverse. Supporta diverse risoluzioni cromatiche (**CMYK**, **RGB**, scala di grigio) e consente di utilizzare un algoritmo di compressione *lossless*.

Le immagini per il *Web* sono tipicamente destinate alla visualizzazione sul monitor, dunque memorizzate a 72 dpi. Ciò nonostante le dimensioni di *file* non compressi o poco compressi sono tali da rendere difficile, e a volte impossibile, la trasmissione di oggetti di queste dimensioni attraverso la rete. Per questo motivo i principali formati per le immagini su *Web* sono memorizzate tramite formati (**GIF**, **PNG** e **JPEG**) che realizzano forme di compressioni molto efficaci.

Compressione

In generale la **compressione** è una particolare operazione di codifica, nella quale l'obiettivo è quello di generare un messaggio codificato che abbia una dimensione minore del messaggio sorgente. Gli algoritmi di compressione, che introduciamo trattando le immagini ma approfondiremo nelle pagine che riguardano audio e video, possono essere classificati come:

- **Lossless** (senza perdita): la compressione è reversibile, ovvero dall'informazione compressa è possibile ricostruire esattamente l'informazione originale.
- **Lossy** (con perdita): la compressione è irreversibile poiché non è più possibile ricostruire esattamente l'informazione originale. In questo caso viene codificata una quantità minore di informazioni e questo permette di ottenere rapporti di compressione nettamente maggiori rispetto a quelli che si hanno con la compressione *lossless*. D'altra parte, la perdita di dettagli che si ha con la compressione *lossy* può non essere percepibile.

Le meccaniche di compressione si distinguono anche in:

- *source encoding*: l'algoritmo di compressione tiene conto del tipo di informazione digitale, ovvero il processo di compressione trae vantaggio dal conoscere a priori se si tratta di un'immagine, di audio, eccetera.
- *entropy encoding*: si basa su tecniche matematiche, tratte dalla teoria dei codici e dell'informazione, e permette di ottenere la compressione di un flusso di dati binario senza conoscere il tipo di informazione in esso contenuto.

Formati per il Web

In particolare i formati tipici per le immagini inserite nelle pagine *Web* sono tre:

- **JPEG** (*Joint Photographic Experts Group*) è un metodo di compressione con perdita di tipo *source encoding* basato sulla conversione dello spazio dei colori. La compressione **JPEG** lavora su immagini *true color* (24 bit per *pixel*) e per immagini con tonalità continue (ovvero per quelle di tipo fotografico) raggiunge facilmente un rapporto di compressione 25:1 senza riduzioni percettibili della qualità.
- **GIF** (*Graphic Interchange Format*): utilizza una compressione di tipo *entropy encoding* e senza perdita che consiste sostanzialmente nell'usare codifiche compresse per sequenze di colori uguali. Questa meccanica rende questo formato particolarmente adatto alla codifica di icone e immagini simboliche e contemporaneamente poco adatto a immagini di tipo fotografico. Usa una risoluzione dei colori basata su *palette* (ovvero 256 colori, un *byte* per ogni *dot*). Lo *standard GIF89* consente la creazione di immagini con trasparenza definendo un valore della *palette* come trasparente. Lo stesso formato consente la costruzione di semplici immagini in movimento. Le immagini in formato **GIF** sono a volte utilizzate per realizzare semplici animazioni. Questo è possibile perché il formato **GIF89** prevede la presenza del *Control Extension Block* che consente di definire sequenze di immagini (immagini multiple) che possono essere visualizzate come una animazione. L'immagine animata va dunque realizzata mediante un paziente lavoro di costruzione delle singole figure che compongono la sequenza. È possibile approfondire le tecniche di costruzione delle **GIF** animate attraverso i *link* suggeriti dalla **Bibliografia**. Le immagini **GIF** possono essere anche interlacciate, ovvero visualizzate in modo non sequenziale. Una immagine **interlacciata** viene visualizzata progressivamente aparendo all'inizio con una qualità bassa, e migliorando gradualmente fino a raggiungere la completa definizione.
- **PNG** (*Portable Network Graphics*, si pronuncia *ping*), è un formato nato con lo scopo di fornire una alternativa a **GIF** nella memorizzazione *lossless* di immagini per il *Web*. Il formato **GIF** utilizza un algoritmo di compressione proprietario per cui il *software* che lo incorpora deve sottostare al pagamento dei diritti. **PNG** è invece gratuito e inoltre presenta altri vantaggi tra cui la possibilità di usare colori *True Color RGB* e l'uso di meccaniche di compressione *lossless* ma più efficaci rispetto a quelle utilizzate da **GIF**.

Markup per le immagini

Per inserire una immagine all'interno della pagina *HTML* viene tipicamente utilizzato il *tag* `IMG` e specificando l'*URL* del *file* in cui è contenuta l'immagine attraverso l'attributo obbligatorio `SRC`. Per ottenere contenuti immagine comprensibili e navigabili anche quando l'utente ha problemi, fisici o contestuali, a visualizzare l'immagine è opportuno fornire una **descrizione testuale alternativa** tramite l'attributo `ALT`. Le specifiche *HTML 4.01* prevedono di definire `ALT` come attributo obbligatorio.

La porzione di codice che segue inserisce nella pagina l'immagine di esempio:

```
<IMG SRC="fioril.jpg" ALT="Immagine di esempio: api su un fiore">
```

Altri attributi (tutti facoltativi) per `IMG` sono:

- `LONGDESC`: collegamento al *file* che contiene una descrizione testuale estesa dell'immagine che viene a volte usata per integrare il contenuto informativo offerto da `ALT`.
- `NAME`: specifica il nome dell'immagine (viene usato soprattutto per l'uso da parte di *script*).
- `HEIGHT`: ridefinisce l'altezza dell'immagine.
- `WIDTH`: ridefinisce la larghezza dell'immagine.
- `USEMAP`: specifica l'uso di mappe dal lato *client*.
- `ISMAP`: specifica l'uso di mappe dal lato *server*.

Gli attributi delle immagini che servono a definirne il *layout* (formattazione e allineamento) sono deprecati dalla versione 4.01 a favore dell'uso dei fogli di stile anche per la formattazione delle immagini. Le ultime versioni di `HTML` definiscono un *tag* per l'inserimento di oggetti esterni, `OBJECT`, che dovrebbe essere usato anche per le immagini. La porzione di codice che segue è equivalente a quella indicata sopra e utilizza `OBJECT` per inserire l'immagine. Si noti come la descrizione testuale alternativa (che nel *tag* `IMG` è gestita attraverso l'uso degli attributi `ALT` e `LONGDESC`) venga inserita in questo caso tra `<OBJECT>` e `</OBJECT>` e possa essere formattata.

```
<OBJECT data="fioril.jpg" type="image/jpeg">
```

Immagine di esempio: api su un fiore

```
</OBJECT>
```

Lo stesso *tag* può essere usato per inserire qualunque tipo di oggetto esterno incluse le *applet Java* e le animazioni *Flash*. A questo scopo lo introdurremo meglio trattando i formati multimediali, audio e video, nella prossima parte dell'introduzione.

Grafica vettoriale 2D e 3D

Le immagini vettoriali sono rappresentazioni visive costruite da descrizioni matematiche di uno o più elementi grafici che possono essere costruite aggiungendo o rimuovendo forme, o applicando trasformazioni geometriche.

Quando le funzioni che compongono la figura sono definite nel piano, si ottengono figure vettoriali bidimensionali (2D) mentre se sono definite nello spazio, si ottengono figure vettoriali tridimensionali (3D). Passare dalla grafica vettoriale in 2D a quella in 3D comporta un ovvio aumento della complessità matematica del problema. La differenza sostanziale risiede però nel fatto che mentre nella grafica 2D le riproduzioni degli oggetti sono comunque astrazioni di tipo simbolico, con la grafica 3D si producono modelli che mirano a rappresentare gli oggetti reali in modo efficace.

I più frequenti ambienti d'uso della grafica vettoriale bidimensionale sono i seguenti:

- La produzione grafica: la creazione di loghi e immagini che devono essere frequentemente ridimensionate e adattate a differenti contesti è realizzata tipicamente con formati vettoriali che si prestano a questo tipo di operazioni.

- La tipografia: i caratteri (ovvero i *font*) sono realizzati con descrizioni vettoriali che consentono di specificare la forma senza preoccuparsi delle dimensioni, che vengono cambiate da una semplice trasformazione matematica.
- La produzione di *content Web* animato: sempre più spesso le pagine *Web* includono oggetti prodotti con strumenti di animazione 2D vettoriale.

Le principali applicazioni di grafica vettoriale tridimensionale sono invece:

- la progettazione industriale e la progettazione edile e urbanistica che vengono realizzate mediante l'uso di *software CAD (Computer Aided Design)*.
- La realizzazione di animazioni 3D e di effetti speciali per l'industria cinematografica.
- I videogiochi.
- La simulazione scientifica e la realtà virtuale.

Animazione vettoriale

L'animazione vettoriale 2D viene usata sempre più frequentemente per costruire siti ad alta interattività o semplicemente per rendere più accattivante il *layout* delle pagine. Un *tool* di particolare successo che realizza animazioni di questo tipo è *Macromedia Flash*. In questa sede ricordiamo che le animazioni di questo tipo costituiscono una tipica barriera all'accesso ai siti e sono dunque in generale poco adatte ad essere ospitate in un sito scolastico, a meno di utilizzare un insieme di opportuni meccanismi che consentano il rispetto dei criteri di accessibilità.

Le animazioni 2D vettoriali sono tipicamente realizzate utilizzando due tecniche base di animazione:

- *cel-like animation*, in cui lo sfondo resta fisso e si muovono solo alcuni elementi posti in primo piano. Questa metodica è realizzata utilizzando diversi livelli di costruzione dell'immagine.
- *sprite*, immagini che si muovono localmente e vengono spostate sullo schermo, indipendentemente dalle altre immagini presenti, dando l'idea di moto complessivo. Questa metodica è realizzata consentendo di costruire elementi in movimento autonomo e offrendo sistemi che li integrano tra di loro.
- *Key frame*: l'animazione è specificata attraverso i punti chiave dell'animazione (*key frame*) in cui i vari elementi cambiano stato. Il programma calcola i passaggi intermedi tra un punto chiave e l'altro provvedendo ad applicare agli elementi le trasformazioni necessarie a passare da un *key frame* al *key frame* successivo. In questo modo il progettista deve solo definire gli stati principali dell'animazione e il programma calcola automaticamente tutti gli stati intermedi.

Prodotti

Quello che segue è un elenco dei principali prodotti disponibili per la grafica (vettoriale e *raster*). L'elenco non vuole essere esaustivo, ma solo citare le applicazioni più diffuse indicando per ciascuna la tipologia (*raster* o vettoriale) e il proprio sito di riferimento. Le informazioni contenute in questa pagina sono soggette a veloce obsolescenza e sono state verificate l'ultima volta al momento della pubblicazione.

Prodotto	Grafica	Sito
<i>Adobe Photoshop</i>	Bitmap	http://www.adobe.com/products/photoshop/

<i>Macromedia Fireworks</i>	Vettoriale e Bitmap	http://www.macromedia.com/software/fireworks/
<i>Corel Draw Graphic suite</i>	Vettoriale e Bitmap	http://www.corel.com
<i>Jasc Paintshoppro</i>	Bitmap	http://www.jasc.com/products/paintshoppro/
<i>Corel Draw</i>	Vettoriale	http://www.corel.com
<i>Macromedia Freehand</i>	Vettoriale	http://www.macromedia.com/software/freehand/
Conclusioni		

Scopo di questa trattazione è stato quello di introdurre brevemente i principali *media* digitali e le loro caratteristiche più salienti. Sono stati inoltre presentati due *media* visuali, le immagini e le animazioni, delle quali sono state delineate le caratteristiche fondamentali.

Tra queste uno dei fattori più rilevanti è ovviamente quello dimensionale. Gli elementi multimediali sono in generale accattivanti e capaci di trasmettere efficacemente contenuti informativi complessi. Per contro però la loro digitalizzazione produce tipicamente *file* di grandi dimensioni che devono essere fortemente compressi per poter essere distribuiti sulla rete. Questo fenomeno, che abbiamo introdotto facendo riferimento alle immagini, sarà ovviamente ancora più accentuato nel caso di *media* continui (come l'audio e il video) in cui la dimensione in *byte* del *file* dipende anche dalla durata in secondi del *media*. La prossima sezione dell'introduzione affronterà appunto questi due *media* digitali introducendo i principali formati e le meccaniche di compressione.

In questa sezione è stato anche introdotto il *markup* necessario ad integrare nei siti *Web media* diversi dal testo. L'inserimento di elementi multimediali nella pagina, se non correttamente attuata, rischia di introdurre barriere d'accesso per chi, per ragioni fisiche o contestuali, non riesce ad utilizzare tutti gli strumenti di *input* e di *output* disponibili in un PC, ovvero non riesce a fruire di tutti i *media* messi a disposizione. Il fatto che un sito risulti accessibile a tutti è particolarmente critico quando, come nel caso delle scuole, le pagine sono strumento per il compimento di un pubblico servizio. Sui principali criteri per la realizzazione di siti accessibili è disponibile un **approfondimento**.

È inoltre disponibile un **approfondimento** sul trattamento e il ritocco delle immagini digitali. Ulteriori approfondimenti sono possibili attraverso il materiale *on-line* suggerito dalla **Bibliografia**.