

# UNIVERSITA' DEGLI STUDI DI CATANIA

DIPARTIMENTO DI MATEMATICA E INFORMATICA

Corso di Laurea Magistrale in Informatica

Federico Petrina 1000012437

## **Applicazione algoritmi di Dithering per le immagini digitali**

Progetto Multimedia

Docenti:

Prof. Dario Allegra

Prof. Filippo Stanco

### 1. Introduzione (cos'è il dithering)

Il Dithering è una tecnica che consente ad una bitmap di visualizzare più colori di quanto sia possibile. In particolare se la bitmap permette solo il bianco e nero, il dithering può essere utilizzato per riprodurre un'illusione di grigio, se invece consente una tavolozza di colori, il dithering può produrre un'illusione facendo percepire nuovi colori della tavolozza che in realtà non sono presenti.

### 2. Stato dell'arte del problema

Nell'elaborazione delle immagini tale tecnica viene applicata in computer grafica per creare l'illusione della profondità di colore in immagini dotate di una tavolozza limitata (quantizzazione del colore). In un'immagine sottoposta a dithering, i colori non disponibili vengono approssimati dalla distribuzione dei pixel colorati con le tinte disponibili. In tal modo l'occhio umano percepisce la diffusione come un miscuglio dei colori.

Tuttavia presenta anche degli svantaggi poiché ridurre la profondità del colore di un'immagine spesso causa effetti collaterali indesiderati. Ad esempio se l'immagine originale è una fotografia i colori saranno probabilmente migliaia e limitarne il numero farà perdere qualità all'immagine. Inoltre le immagini trattate con il dithering, in particolare quelle in cui vengono usati pochi colori, perdono in termini di comprimibilità e possono spesso apparire granulari o composte da puntini.

### 3. Algoritmi applicati

Esistono molti algoritmi creati per l'applicazione del dithering. Uno dei primi, ed ancora tra i più popolari, è l'algoritmo di **Floyd-Steinberg**, sviluppato nel 1975. Uno dei punti di forza di questo algoritmo è la sua capacità di minimizzare gli artefatti visivi attraverso un processo di diffusione dell'errore; solitamente produce immagini più simili all'originale rispetto a quanto fatto da altri algoritmi simili. Tale metodo tende a produrre risultati efficaci con relativamente pochi artefatti ripetuti. Lo svantaggio principale consiste nel fatto di essere applicabile relativamente ad alta intensità di processore e quindi può essere lento per i computer su grandi aree.

Questo algoritmo compie il dithering diffondendo l'errore di quantizzazione di un pixel ai pixel vicini. Più specificamente, nell'algoritmo proposto, l'errore di un pixel viene diffuso ai pixel circostanti in queste proporzioni:

7/16 dell'errore viene sommato al pixel a destra, 3/16 al pixel in basso a sinistra, 5/16 al pixel sottostante, e 1/16 al pixel in basso a destra.

$$\frac{1}{16} \begin{bmatrix} - & \# & 7 \\ 3 & 5 & 1 \end{bmatrix}$$

L'algoritmo esamina l'immagine da sinistra a destra e dall'alto verso il basso, quantizzando i valori dei pixel uno ad uno, trasferendo l'errore di quantizzazione di ogni pixel a quelli adiacenti, senza però coinvolgere quelli già quantizzati. Quindi se un certo numero di pixel è già stato arrotondato per difetto, è lecito attendersi che il prossimo pixel sia arrotondato per eccesso, per cui la media della quantizzazione dell'errore è vicina a zero.

I coefficienti di diffusione hanno la proprietà che se i valori dei colori di partenza della tonalità del pixel sono esattamente uguali il risultato del dithering sarà un motivo a scacchiera.

Un altro algoritmo è il **Jarvis Dithering** esso è un metodo simile che è stato definito “errore medio minimizzato”, esso utilizza un kernel più grande rispetto a Floyd-Steinberg diffondendo l'errore anche ai pixel un passo più avanti. Tale algoritmo di dithering tende a produrre risultati efficaci con relativamente pochi artefatti ripetuti, inoltre riesce a produrre modelli più grossolani rispetto al più noto metodo di dithering Floyd-Steinberg, ma con meno artefatti visivi.

Tuttavia è più lento perché distribuisce errori tra 12 pixel vicini anziché 4.

$$\frac{1}{48} \begin{bmatrix} - & - & \# & 7 & 5 \\ 3 & 5 & 7 & 5 & 3 \\ 1 & 3 & 5 & 3 & 1 \end{bmatrix}$$

Jarvis Dithering è strettamente correlato a Stucki Dithering, ma è leggermente più ad alta intensità di processore e le immagini prodotte con Jarvis Dithering tendono ad essere leggermente più sfocate rispetto a quelle prodotte con Stucki Dithering. Il principale svantaggio di tale algoritmo consiste nell'essere applicabile esclusivamente ad alta intensità di processore, e quindi può essere lento per il computer da eseguire su grandi aree.

#### 4. Metriche utilizzate

**MSE:** serve a stimare l'errore quadratico medio tra due immagini, più tale indice è basso minore è la differenza fra le immagini.

$$MSE = \frac{\sum_{i=1}^n (x_i - \hat{x}_i)^2}{n}$$

**PSNR:** misura adottata per valutare la qualità di un'immagine compressa rispetto all'originale; maggiore è il suo valore maggiore sarà la somiglianza con l'originale.

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

**SSIM:** Al fine di valutare la qualità dell'immagine, questa formula viene solitamente applicata solo ai luma (ovvero la luminosità in un'immagine, la parte "in bianco e nero" o acromatica), sebbene possa essere applicata anche ai valori di colore (ad es. RGB) o cromatici (ad es. YCbCr). L'indice SSIM risultante ha un valore decimale compreso tra -1 e 1 e il valore 1 è raggiungibile solo nel caso di due serie identiche di dati e pertanto indica una perfetta somiglianza strutturale. Un valore pari a 0 non indica alcuna somiglianza strutturale.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

**MSE<sub>v</sub>**: Metrica di visibilità dei mezzi toni utilizzata per misurare la distorsione osservata da uno spettatore umano tra un'immagine X di livello di grigio originale e il suo semitono binario B. In particolare una di queste è l'MSE<sub>v</sub> ed è definita come segue:

$$\text{MSE}_v = \frac{1}{N \times N} \|hvs(X, vd, dpi) - hvs(B, vd, dpi)\|^2$$

dove **hvs** è la funzione di filtro HVS (human visual system), **vd** è la distanza di visualizzazione in pollici e **dpi** è la risoluzione della stampante. Nelle simulazioni la distanza di visualizzazione varia da 20 a 80 pollici ed è stata considerata una risoluzione della stampante di 600 dpi.

## 5. Esperimenti effettuati e risultati

Per quanto riguarda gli algoritmi sono stati applicati il Floyd-Steinberg dithering e il Jarvis dithering ad un dataset di 30 immagini in formato PNG (sia a colori che a scala di grigio) di differenti dimensioni. Di seguito è mostrato un esempio:



*Originale Vs Floyd-Steinberg*



*Originale Vs Jarvis*

Il risparmio in termini di bit per colore dopo l'applicazione di questi algoritmi poiché prima si hanno 24 bit (con circa 16 milioni di colori disponibili) e dopo il dithering si hanno a disposizione solo N colori è pari al  $\lceil (\log_2 N) \rceil$ .

In seguito per valutare la somiglianza di questi output con i rispettivi originali si sono applicate prima le metriche oggettive sopra citate, seppur poco affidabili nella veridicità dei valori ottenuti. Dei risultati è stata calcolata la media in termini di MSE, PSNR e SSIM (i valori di ogni singola immagine sono riportati nel file Excel in allegato).

Successivamente, per ottenere riscontri più precisi, si è tentato di applicare la metrica soggettiva nota come MSEv, tuttavia data l'assenza della "hvs filter function", necessaria per la realizzazione ottimale, si è proceduto operando nella seguente maniera:

Uno script mostra in ordine casuale ad un osservatore due immagini che possono essere di tre tipi:

1. Originale & Originale
2. Originale & e la sua corrispondente con il Floyd-Steinberg dithering
3. Originale & e la sua corrispondente con il Jarvis dithering

Ovviamente l'osservatore non sa in quale dei tre casi si trova e ad esso viene chiesto di esprimere un giudizio da 1 a 10 sulla somiglianza tra le due immagini, dove il valore 1 rappresenta la totale differenza ed il valore 10 la totale somiglianza. Il test è stato effettuato su 3 soggetti senza particolari problemi visivi (al massimo muniti di occhiali) e ripetuto più volte su diverse immagini e ad una distanza di visualizzazione tra 50 cm (corrispondente a 20 pollici) e 2 metri (corrispondente a 80 pollici). Nel file Excel allegato sono riportati i test effettuati sui soggetti con dodici immagini a distanza di circa 50 cm, 1 metro e 2 metri, infine dei valori ottenuti è stata fatta una media sia tra le varie distanze sia tra i voti ottenuti applicando l'algoritmo di Floyd e di Jarvis.

Dai valori riscontrati dagli esiti delle due tipologie di metriche si evince che l'applicazione di algoritmi di dithering sulle immagini riporta in media valori matematici che evidenziano molta differenza tra gli originali, tuttavia per un osservatore posto ad una certa distanza questa differenza non è percepita.

Quindi soprattutto nel caso del Floyd dithering (che risulta percettivamente migliore rispetto al Jarvis) l'originale e la sua corrispondente a cui è stato applicato l'algoritmo risulteranno pressoché identiche.

	MSE(Floyd)	PSNR(Floyd)	SSIM(Floyd)	MSE(Jarvis)	PSNR(Jarvis)	SSIM(Jarvis)
<b>Media</b>	68,52	29,79	0,61	67,86	29,83	0,68

*Valori arrotondati delle metriche oggettive*

Distanza	50 cm	1 m	2 m
<b>Media</b>	8,15/10	8,13/10	8,36/10

*Valori arrotondati della metrica soggettiva*



Algoritmo	Floyd	Jarvis
Media	8,15/10	8,13/10

*Valori arrotondati dei voti ottenuti*

## 6. Undithering

Consiste nel processo di ricostruzione dell'immagine in scala di grigio o a colori dall'immagine a cui è stato applicato il dithering. Come descritto nell'articolo di riferimento [1] viene applicato un metodo di distacco utilizzando il filtro lineare seguito dalla diffusione anisotropica che porta il vantaggio di levigare e migliorare i bordi. I parametri statistici del primo ordine, i parametri statistici del secondo ordine, l'errore quadratico medio (MSE) tra l'immagine ricostruita e l'immagine originale prima del dithering e il rapporto segnale / rumore (PSNR) sono valutati ad ogni fase di diffusione.

Si è quindi proceduto tentando di replicare la suddetta tecnica sulle immagini del dataset precedente a cui sono stati applicati i due algoritmi di dithering, al fine di ottenere un'immagine con meno granularità e più simile all'originale.

Per realizzare ciò è stata utile una repository su github [2] la quale usa un algoritmo che applica un filtro per rimuovere il dithering (come il Floyd-Steinberg) dalle immagine palettizzate. Tale metodo analizza la tavolozza delle immagini per trovare la soglia di sfocatura ottimale. Per due pixel adiacenti se la tavolozza ha un colore compreso tra i colori di questi due pixel, si presume che sia un bordo.

In particolare data in input immagine a 8-bit (PNG8) a cui è stato applicato il dithering, restituirà un'immagine a 24-bit (PNG24 true color).

Ecco due esempi di Undithering, rispettivamente su un'immagine in scala di grigi e su una a colori:



*Dithered image*



Undithered image



*Dithered image*



Undithered image

In conclusione i risultati degli esperimenti effettuati mostrano che l'immagine ricostruita non è nitida come l'immagine prima del dithering ma la sua granularità è stata visivamente ridotta, a dimostrazione del fatto che un gran numero di valori di pixel viene riprodotto con riferimento a quelli dell'immagine originale prima del dithering.

## 7. Riferimenti

1. V.Asha “Undithering using linear filtering and non-linear diffusion techniques”: <https://arxiv.org/ftp/arxiv/papers/1108/1108.2475.pdf>
2. Undithering repository: <https://lib.rs/crates/undither>
3. Algoritmo Floyd-Steinberg;  
[https://it.wikipedia.org/wiki/Algoritmo\\_di\\_Floyd-Steinberg](https://it.wikipedia.org/wiki/Algoritmo_di_Floyd-Steinberg)
4. Algoritmo Jarvis: [https://en.wikipedia.org/wiki/Error\\_diffusion](https://en.wikipedia.org/wiki/Error_diffusion)
5. IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL.19, NO. 7  
JULY 2010: “Green Noise Digital Halftoning With Multiscale Error  
Diffusion”:  
[https://www.researchgate.net/publication/224122244\\_Green\\_Noise\\_Digital\\_Halftoning\\_With\\_Multiscale\\_Error\\_Diffusion](https://www.researchgate.net/publication/224122244_Green_Noise_Digital_Halftoning_With_Multiscale_Error_Diffusion)