**Università Degli Studi di Salerno**

**Dipartimento di informatica**

*Corso di Laurea Magistrale Cloud Computing*



**Unione Dei Progetti Di:**

* Compressione Percettivo Semantica Dell’Immagine Usando Reti Neurale Di Convoluzione
* Compressione Di Immagini Composite

|  |  |
| --- | --- |
| **Docente**  Prof. Bruno Carpentieri  Prof. Raffaele Pizzolante | **Gruppo: Negan**  Capaldo Giuseppe 0522500498  Donadio Raffaele 0522500423 |

Indice

[1. Introduzione 4](#_Toc506899698)

[2. Un primo tentativo 6](#_Toc506899699)

[3. Simulazione 30](#_Toc506899700)

[4. Testing 34](#_Toc506899701)

[5. Conclusioni 49](#_Toc506899702)

# 

# Introduzione

I segnali analogici (musica, voce, tv e film) sono più difficili da memorizzare e da trasmettere su lunghe distanze a differenza dei digitali. Lo svantaggio principale che riguarda le versioni digitali dei segnali è il fatto che richiedono più bits. Per ridurre i costi di memorizzazione e trasmissione, si adoperano tecniche di compressione, il cui scopo è minimizzare il numero di bits necessari per ottenere una versione accettabile del segnale sorgente originale.

Però tutte le tecniche adoperate negli ultimi tempi sono state tecniche che non hanno tenuto conto di cosa bisognasse comprimere; questo ha portato all’immissione di troppi errori all’interno dell’immagine che si traduce in una pessima qualità visiva.

Il documento che ci è stato consegnato e che abbiamo messo sotto esame è il paper Semantic Perceptual Image Compression Using Deep Convolution Networks di Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo and James Storer della BrandeisUniversity. Nel suddetto si parla dell’utilizzo delle Reti Neurali Convoluzionali (CNN) e di come i recenti progressi nella potenza di calcolo, insieme alla disponibilità di ampi set di dati sull'addestramento, hanno aumentato l'interesse nell'applicazione di CNN’s di deep learning per affrontare il riconoscimento dell'immagine e le attività di elaborazione delle immagini.

Questo paper riporta anche il link di una sua prima implementazione già presente su github;

in cui vengono usati:

1. Tensorflow
2. Numpy
3. Pandas
4. Python PIL
5. PythonSKimage

Inoltre vengono suggeriti:

1. Imagemagick (per operazioni di immagine più veloci)
2. VQMT (per ottenere metriche di confronto delle immagini)

Tutto questo progetto è fondato su Python e sulle CNN, ma purtroppo nessuno del gruppo ha avuto la possibilità di studiare l’ambito dell'intelligenza artificiale e quindi non avendo le giuste competenze per capirlo fino in fondo è stato preso in considerazione come una black-box.

Il progetto in questione si prefigge l’obiettivo di rendere jpeg content-aware, ovvero pur applicando sempre lo stesso algoritmo per compressore e decompressore, si fa in modo di cambiare dinamicamente la qualità della compressione appunto usando la CNN.

Infatti alla fine del paper in questione si spiega che l’unico punto su cui si può migliorare ciò che loro stesso hanno fatto è la CNN migliorando in tutti i suoi aspetti le performance.

Adesso invece prendiamo in considerazione il secondo progetto che ci è stato fornito dal professore:

il progetto è stato realizzato dai nostri colleghi Aiello Michele, Bisogno Giuseppe, Manzo Gerardo, Paolella Vincenzo nell’anno Accademico 2007/2008.

In questo secondo progetto si parla della compressione dell’immagine mediante la separazione in più livelli della stessa, facendo sì che si possano usare livelli di compressione o, addirittura, tipi di compressione diversi per ogni livello facendo sì che al riassemblamento del tutto non si perda in qualità ma solo in dimensione.

I livelli che loro generano sono 3:

Background, Foreground e Maschera.

Il Background è il livello dei pixel che indica oggetti di fondo a cui l’occhio umano presterà meno attenzione.

Il Foreground è il livello dei pixel che indica gli oggetti in primo piano, quelli a cui l’occhio umano presterà più attenzione.

La Maschera è il livello che ci permette di estrarre i livelli sopra-citati dall’immagine originale.

# Un primo tentativo

Inizialmente siamo partiti con l’utilizzare il progetto dei colleghi, per separare l’immagine nei 3 livelli che la compongono e quindi usiamo una compressione molto forte sul livello di Background e una compressione lieve sul Foreground così da ottenere, alla ricomposizione dell’immagine, un ottimo valore di compressione senza però perderne in qualità. O almeno queste erano le nostre premesse.

*Vedendo che i primi risultati non rispecchiavano le aspettative abbiamo capito che i colleghi invertono il Foreground dell’immagine con il Background. Facendo numerosi test si è capito che Il problema fondamentale del progetto ereditato è che tutta la struttura software si basa su file jar per la compressione, il quale non lavora come dovrebbe, infatti comprimendo la stessa immagini in due sessioni nelle quali venivano assegnati diversi parametri di qualità, il PSNR risultava sempre uguale.*

A questo punto confrontandoci anche con il professore sulla natura fallimentare di tutti questi tentativi, abbiamo deciso di cambiare algoritmo di segmentazione passando da suddivisione in Foreground e Background ad algoritmi del tipo superpixel. Inoltre abbiamo creato partendo dalle basi un progetto basato su linguaggio Python.

Quindi abbiamo creato un progetto la cui idea di base consiste nel prendere in input un’immagine e applicarvi un algoritmo di segmentazione, ottenendo un’immagine segmentata secondo la logica del singolo algoritmo di segmentazione.

Una volta ottenuto i frammenti si procede con la selezione dei frammenti più rilevanti visivamente parlando (**simulando la rete neurale**).

**Data la seguente immagine:**



I frammenti più rilevanti potrebbero essere i frammenti che contengono il viso dell’astronauta e la tuta spaziale.

Una volta identificati i frammenti più importanti si procede con la compressione dei singoli , usando una qualità di compressione maggiore per i frammenti importanti e minore altrimenti.

Finita la fase di compressione si fa un merge di tutti i frammenti per riottenere l’immagine completa.

**Il processo può essere rappresentato tramite il seguente modello:**



1. **Estrazione delle maschere**

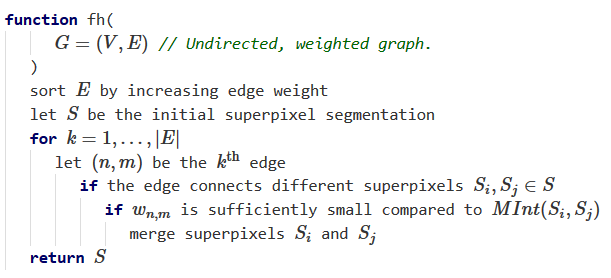
L’estrazione delle maschere avviene mediante l’algoritmo di segmentazione.

Per la segmentazione abbiamo usato la tecnica del superpixel precisamente ci siamo concentrati sulle varianti Felzenswalb & Huttenlocher e SLIC.

La variante Felzenswalb funziona considerata un'immagine e il grafico con nodi corrispondenti a pixel e bordi tra pixel adiacenti. Ad ogni bordo viene assegnato un peso . Quindi, per i sottoinsiemidefiniamo per essere il minimum spanning tree di ***A*** e

.

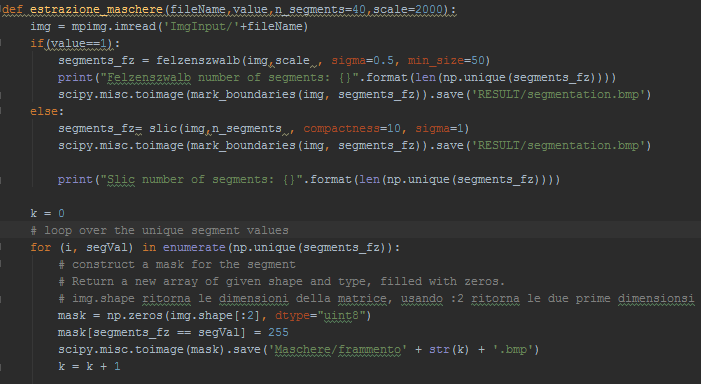
Dove τ è un parametro di soglia eè chiamato la differenza interna minima tra i componenti *A* e *B*. Partendo da una prima segmentazione di superpixel in cui ogni pixel forma il proprio superpixel, l'algoritmo elabora tutti i bordi in ordine di aumento del peso del bordo. Ogni volta che un bordo collega due superpixel diversi, questi vengono uniti se il peso del bordo è piccolo rispetto alla differenza interna minima.



Mentre l'approccio SLIC esegue un clustering locale di pixel nello spazio 5-D definito dai valori L, a, b dello spazio colore CIELAB e delle coordinate x, y dei pixel. Ha una diversa misurazione della distanza che consente compattezza e regolarità nelle forme dei superpixel e può essere utilizzata su immagini in scala di grigi e immagini a colori. SLIC genera superpixel raggruppando i pixel in base alla somiglianza e alla vicinanza dei colori nel piano dell'immagine. Per il clustering viene utilizzato uno spazio **[labxy]** a5 dimensioni.Lo spazio colore CIELAB è considerato uniforme perpetuamente per piccole distanze di colore.Non è consigliabile utilizzare semplicemente la distanza euclidea nello spazio 5D e quindi gli autori hanno introdotto una nuova misura di distanza che considera la dimensione dei superpixel.Si inizia campionando K centri di cluster regolarmente distanziati e spostandoli in posizioni di seme corrispondenti alla posizione di gradiente più bassa in un quartiere 3 × 3.Questo è fatto per evitare di metterli a un bordo e ridurre le possibilità di scegliere un pixel rumoroso.I gradienti di immagine sono calcolati come

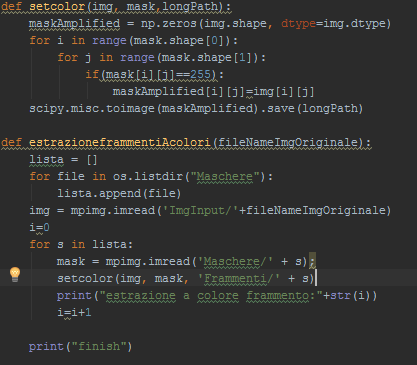
dove è il vettore lab corrispondente al pixel in posizione e è la norma L2. Questo tiene conto sia delle informazioni sul colore che sull'intensità. Ogni pixel dell'immagine è associato al centro del cluster più vicino la cui area di ricerca si sovrappone a questo pixel. Dopo che tutti i pixel sono associati al centro del cluster più vicino, un nuovo centro viene calcolato come vettore lab. Medio di tutti i pixel appartenenti al cluster. Al termine di questo processo, possono rimanere alcune etichette vaganti, ovvero pochi pixel in prossimità di un segmento più grande con la stessa etichetta ma non collegato ad esso. Mette in atto la connettività nell'ultimo passo dell'algoritmo rielaborando i segmenti disgiunti con le etichette del cluster più grande vicino.

**Ecco lo script python per l’estrazione delle maschere**



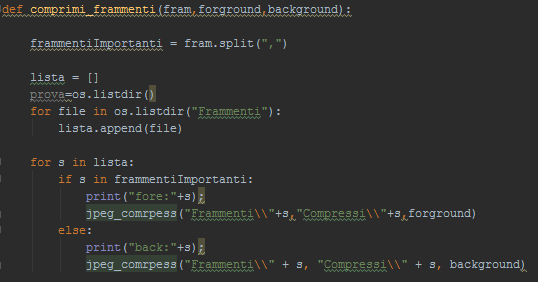
1. **Estrazione dei frammenti**

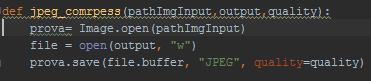
A partire dall’immagine di input e dalla maschera estraiamo i pezzi dell’immagine così da ottenere i frammenti di immagine a colore e in formato bitmap in modo da poter scegliere quali comprimere di più e quali di meno e lo facciamo con una funzione che richiama l’altra così da ottenere un codice più leggibile



1. **Compressione dei frammenti**

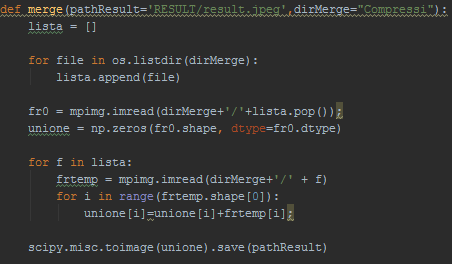
In questa fase si comprimono i frammenti utilizzando JPEG come algoritmo di compressione. Inoltre scelti i frammenti importanti si usa un valore di qualità diverso per quei frammenti.





1. **Unione dei frammenti**

In questa fase ricombiniamo tutti i frammenti compressi in un’unica immagine

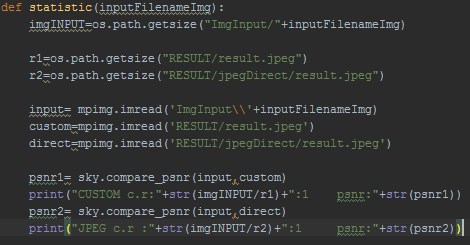


1. **Compressione diretta dell’immagine di input:**

Per ottenere un confronto con l’applicazione del semplice JPEG senza tutti i passaggi sopra-citati chiamiamo la funzione jpeg\_compress con i dovuti parametri.

1. **Calcolo dei Compression Ratio e dei PSNR:**

Inoltre per ottenere dei dati di confronto abbiamo creato una funzione che ci estrae alcuni parametri di confronto qualità e compressione.



**RISULTATI**

|  |
| --- |
| https://lh6.googleusercontent.com/Z7kNrS3halW0ovir_5vzBYUcM65HxCwM-7rTRsEHb4nbVj5aPXPtLdDpg5Fej2-1yLExwTDmg9LuXm5rkt2FIrHG-o54E5B6kMy78WveOVjTTC_21-98SVClQWxFNaudkOEzhfx2 |
| **Image Segmentation**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | |  |  |   **Most Important Segment Seleceted**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | |  |  |   **TEST1-2**   |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | | https://lh6.googleusercontent.com/Z7kNrS3halW0ovir_5vzBYUcM65HxCwM-7rTRsEHb4nbVj5aPXPtLdDpg5Fej2-1yLExwTDmg9LuXm5rkt2FIrHG-o54E5B6kMy78WveOVjTTC_21-98SVClQWxFNaudkOEzhfx2 |  | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **SLIC** | |  | | --- | | **Quality Compression** | | 76 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 18:1 | 33.2 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 18:1 | 31.1 | | | **Originale** | **JPEG** | **Finale nostro** | | https://lh6.googleusercontent.com/Z7kNrS3halW0ovir_5vzBYUcM65HxCwM-7rTRsEHb4nbVj5aPXPtLdDpg5Fej2-1yLExwTDmg9LuXm5rkt2FIrHG-o54E5B6kMy78WveOVjTTC_21-98SVClQWxFNaudkOEzhfx2 |  | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **Felzenswalb** | |  | | --- | | **Quality Compression** | | 76 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 18:1 | 33.2 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 18:1 | 27.1 | | |
| C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\serena.bmp |
| **Image Segmentation**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp |     **Most Important Segment Seleceted**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | segmentation.bmp |   **TEST 3-4**   |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\serena.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **SLIC** | |  | | --- | | **Quality Compression** | | 76 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 38.8 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 38.7 | | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\serena.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **Felzenswalb** | |  | | --- | | **Quality Compression** | | 76 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 38.8 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 38.6 | | |
| C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\tavolo.bmp |
| **Image Segmentation**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | segmentation.bmp |     **Most Important Segment Seleceted**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp |   **TEST 5-6**   |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\tavolo.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **SLIC** | |  | | --- | | **Quality Compression** | | 94 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 1:1 | 60 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 1:1 | 19 | | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\tavolo.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **Felzenswalb** | |  | | --- | | **Quality Compression** | | 84 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 60 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 8:1 | 20 | | |
| C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\delta.bmp |
| **Image Segmentation**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp |   **Most Important Images Seleceted**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp |   **TEST 6-7**   |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\delta.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **SLIC** | |  | | --- | | **Quality Compression** | | 82 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 23:1 | 33 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 23:1 | 25 | | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\delta.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **Felzenswalb** | |  | | --- | | **Quality Compression** | | 78 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 26:1 | 33 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 26:1 | 26 | | |

|  |
| --- |
| C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\AstronautaB&W.BMP |
| **Image Segmentation**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp |   **Most Important Images Seleceted**   |  |  | | --- | --- | | **SLIC** | **Felzenswalb** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\segmentation.bmp | segmentation.bmp |   **TEST 8-9**   |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\AstronautaB&W.BMP | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **SLIC** | |  | | --- | | **QualityCompression** | | 78 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 6:1 | 38 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 6:1 | 35 | | | **Originale** | **JPEG** | **Finale nostro** | | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\AstronautaB&W.BMP | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | C:\Users\Negan\AppData\Local\Microsoft\Windows\INetCache\Content.Word\result.jpeg | | **Felzenswalb** | |  | | --- | | **Quality Compression** | | 78 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 6:1 | 38 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not Important** | | 100 | 89 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 6:1 | 31 | | |

Una volta effettuato i test, risulta un’anomalia:

* A Parità di Compression Ratio JPEG puro ha un PSNR molto differente rispetto al PSNR misurato sull’immagine ottenuta dal processo di frammentazione e compressione delle ROI.

Questi problemi sono dovuti alla cattiva applicazione di alcune funzioni dei pacchetti usati.

Infatti nel momento in cui si estraggono i frammenti, il singolo frammento è un’immagine con background nero e una piccola sezione di immagine originale ottenuta dal processo di segmentazione, quindi nel momento in cui si va a comprimere il singolo frammento in realtà si comprime un’immagine nera con un piccolo pezzettino di immagine originale.

Quindi il passo successivo sarebbe quello di prendere i frammenti ed eliminare completamente tutte le sezioni nere, cioè sostanzialmente eliminare il background, ma questo processo risulta costoso in termini di computazione poiché questo task andrebbe fatto per ogni singolo frammento, quindi poiché l’obiettivo è simulare La rete neurale abbiamo optato per una strategia molto più semplice ovvero: fare una segmentazione matriciale effettuata da noi in modo da evitare frammenti con background nero.

Inoltre il problema dei risultati non è dovuto solamente al discorso appena fatto ma anche perché il processo descritto precedentemente effettua due compressioni per ottenere l’output del processo di frammentazione e compressione.

Potrebbe sembrare una banalità o semplicemente un errore da parte nostra, e in realtà lo è, ma questo errore ci ha illuminato su un problema serio e non eliminabile.

**Cerchiamo di capire meglio il problema valutando tutto il processo:**

Data **A** un’immagine in input, sia **xi** l’i-esimo frammento ottenuto.

Il passo successivo è quello di comprimere tutti i frammenti e ottenere tanti file jpeg.

Bene fatto questo non si può ottenere un solo file jpeg come output perché l’unica soluzione è quella di modellare in memoria tutti i file jpeg (leggere questi file mediante gli strumenti di python che sostanzialmente consiste nel modellare questa foto come una matrice a n dimensioni) ottenendo delle matrici, effettuare il merge, però a quel punto non posso riportarlo in jpeg altrimenti starei comprimendo 2 volte.

Quindi la soluzione viene mostrata nel capitolo successivo

# Simulazione

Tutto il progetto è incentrato sulla compressione dinamica delle immagini e, come già stato detto, non avendo le competenze di CNN, con l’approvazione del professore, abbiamo optato per un artifizio. Ovvero segmentiamo l’immagine, si selezionano i frammenti più importanti e su quelli verrà effettuato una compressione minore. Sui restanti frammenti dell’immagine si esegue una compressione molto più forte così da conservare il più possibile i dettagli importanti.

Questo secondo tentativo differisce dal primo perché:

* Viene effettuata una segmentazione matriciale ovvero data l’immagine si frammenta in tanti pezzi dove il pezzo generico x, y appartiene alla matrice di overlay creata sull’immagine.

Questo ci consente di ottenere esattamente il frammento senza il background nero, perché come già detto nel capitolo precedente gli strumenti utilizzati ottengono dei frammenti con background nero che potrebbe influenzare anche se non in maniera significativa sulla compressione.

* Per evitare il problema di effettuare la compressione due volte quello che si fa dopo la compressione dei frammenti è:

Modellare in memoria tutti i frammenti ottenendo tante matrici quanti sono i frammenti, fare il merge di tutte queste matrici e creando l’output in formato bmp da cui prendere il risultato visivo del processo di segmentazione e compressione e su questa calcolare il PSNR. Per quanto riguarda il calcolo del compression ratio, non avendo a disposizione un solo file jpeg, la dimensione del file di output si ottiene facendo la somma di tutti i frammenti in formato jpeg.

Quindi se ho 10 frammenti in formato jpeg, la dimensione complessiva è data dalla somma di tutte le dimensioni.

Però questo risultato non è veritiero perché jpeg usa dei metadati per ogni file jpeg ed è chiaro che sommare le dimensioni di ogni frammento introduce errore perché significa sommare tanti metadati di ogni singolo frammento.

Per avere un termine di paragone veritiero quello che si fa è: poiché ci sono delle ridondanze cioè gli header di tutti i frammenti, si può comprimere usando un algoritmo lossless togliendo tutti i metadati(ridondanze).

l’insieme dei frammenti jpeg ottenuti dalla segmentazione vengono compressi ottenendo un unico file .tar.xz.

Il file .tar.xz deriva dall’applicazione tramite libreria prima del compressore tar e poi successivamente del compressore LZMA (Lempel-Ziv-Markov). Quest’ultimo è implementato anche come estensione .7z nel famoso programma LGPL (GNU Lesser General Public License) 7zip.

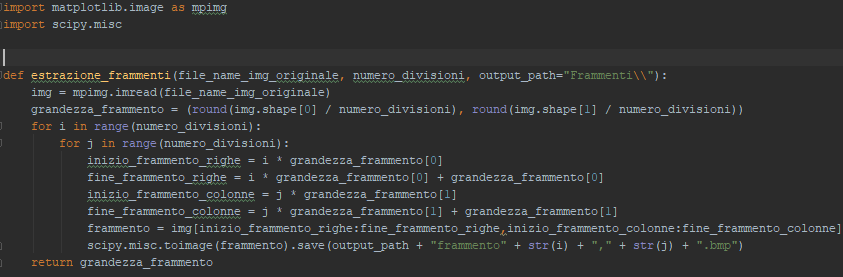
Così facendo abbiamo una dimensione quasi esatta dell’output del processo di segmentazione e compressione.

**Il processo può essere rappresentato tramite il seguente modello:**

.

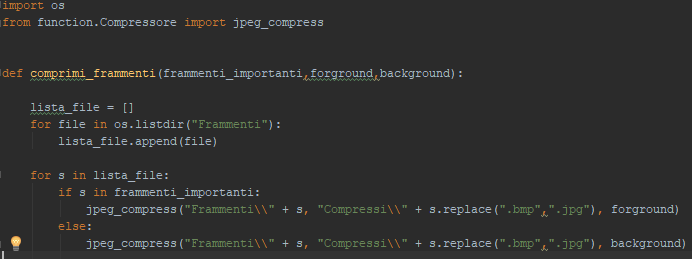
1. **Estrazione dei frammenti**

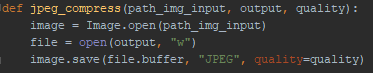
La prima operazione che viene effettuata a partire dall’immagine è la sua suddivisione in sottomatrici di pixel operazione effettuata con il seguente script python



1. **Compressione dei frammenti**

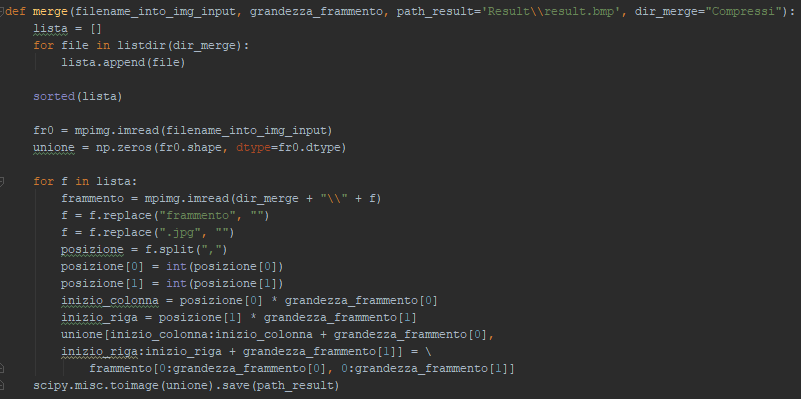
In questa fase si comprimono i frammenti utilizzando JPEG come algoritmo di compressione. Inoltre scelti i frammenti importanti si usa un valore di qualità diverso per quei frammenti.





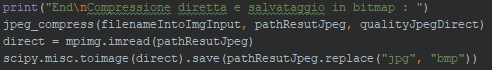
1. **Unione dei frammenti**

In questa fase si comprimono i frammenti utilizzando JPEG come algoritmo di compressione. Inoltre scelti i frammenti importanti si usa un valore di qualità diverso per quei frammenti.



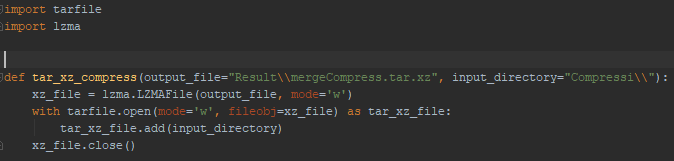
1. **Compressione diretta dell’immagine di input**

Per ottenere un confronto con l’applicazione del semplice JPEG senza tutti i passaggi sopra-citati chiamiamo la funzione jpeg\_compress con i dovuti parametri.



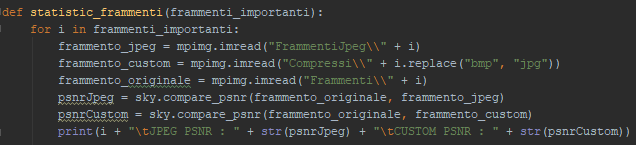
1. **Compressione dei frammenti Jpeg in archivio tar.xz**

Per migliorare il compression ratio della nostra suddivisione e togliere le ridondanze all’interno della cartella di risultato sui metadati di jpeg utilizziamo due compressori lossless tar e LMZA.



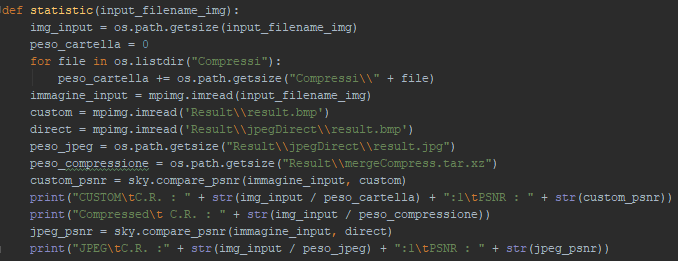
1. **Statistiche dei frammenti**

Per capire se il nostro procedimento porta dei giovamenti alla qualità dei frammenti di interesse li abbiamo confrontati con la controparte Jpeg

****

1. **Statistiche su tutto il file**

Inoltre per ottenere dei dati di confronto abbiamo creato una funzione che ci estrae alcuni parametri di confronto qualità e compressione del risultato e della sua versione ricompressa.



# Testing

|  |
| --- |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 70 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 29.68:1 | 34.52 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 76 | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 22.35:1 | 34.53 | | 29.70:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0,2 | 36.72 | 37.37 | | Frammento 1.0 | 33.65 | 34.18 | | Frammento 1.1 | 31.40 | 31.80 | | Frammento 1.2 | 31.83 | 32.28 | |

|  |
| --- |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 70 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 29.43:1 | 36.7 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 76 | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 30.57:1 | 36.30 | | 32.13:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0,0 | 36.20 | 36.86 | | Frammento 0.2 | 43.50 | 44.50 | | Frammento 1.1 | 33.04 | 33.62 | | Frammento 1.3 | 37.84 | 38.54 | |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 70 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 31,68:1 | 32.6 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 75 | 65 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 26.53:1 | 32.4 | | 32.61:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0,0 | 33.30 | 33.70 | | Frammento 0.1 | 37.27 | 37.60 | | Frammento 1,0 | 30.38 | 30.78 | | Frammento 1,2 | 30.95 | 31.25 | |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 69 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 33.40:1 | 37.30 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 76 | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 23.93:1 | 37.52 | | 33.0:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0.2 | 36.0 | 36.77 | | Frammento 1.2 | 31.94 | 32.84 | | Frammento 2.1 | 35.0 | 35.7 | | Frammento 2.2 | 35.84 | 36.50 | | Frammento 3.1 | 35.25 | 36.1 | | Frammento 3.2 | 37.57 | 38.34 | |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 69 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 22.7:1 | 37.09 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 76 | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 14.73:1 | 37.18 | | 22.07:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 1.1 | 36.39 | 37.48 | | Frammento 1.2 | 34.22 | 35.29 | | Frammento 2.0 | 37.21 | 38.26 | | Frammento 2.1 | 33.43 | 34.44 | | Frammento 2.2 | 35.02 | 36.20 | | Frammento 2.3 | 36.40 | 37.57 | |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 39,68:1 | 36,4 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 90 | 45 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 26.53:1 | 36,3 | | 40.61:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0.1 | 41.33 | 46.65 | | Frammento 0.2 | 41.25 | 46.20 | | Frammento 1,1 | 40.56 | 44.94 | | Frammento 1,2 | 40.38 | 44.56 | | Frammento 2.1 | 37.25 | 42.04 |   **Tutti gli altri frammenti importanti hanno valori simili ….** |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 5.5:1 | 40.1 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 90 | 45 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 2.1:1 | 40.0 | | 5.2:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 1.0 | 39.32 | 44.74 | | Frammento 1.1 | 39.70 | 45.73 | | Frammento 1,2 | 38.90 | 45.18 | | Frammento 1,3 | 38.17 | 44.0.6 | |
|  |
| |  |  |  | | --- | --- | --- | | **Originale** | **JPEG** | **Finale nostro** | |  |  |  | |  | |  | | --- | | **Quality Compression** | | 65 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 28.8:1 | 34.0.1 | | |  |  | | --- | --- | | **Value Of Fragment Important** | **Value Of Fragment Not**  **Important** | | 75 | 60 |  |  |  | | --- | --- | | **Compression Ratio** | **PSNR** | | 19.0:1 | 30.0 | | 28.8:1 |  | | |
| **Examples Local Analysis**   |  |  |  | | --- | --- | --- | | **frammento** | **Jpeg PSNR** | **Custom PSNR** | | Frammento 0.2 | 33.49 | 34.69 | | Frammento 1.2 | 28.90 | 30.62 | |

# Conclusioni

Come possiamo vedere dai risultati abbiamo ottenuto molteplici benefici nel fare tutto questo la prima cosa