UNIVERSITÀ
DI SIENA
1240

Dipartimento di Ingegneria dell'Informazione e
Scienze Matematiche

Corso di laurea in
INGEGNERIA INFORMATICA E DELL'INFORMAZIONE

Affinamento del Riconoscimento Ottico dei Caratteri tramite tecniche di Super Resolution

Relatore:
Prof. Marco Maggini

Candidato:
Pietro Pianigiani

Anno Accademico 2022 - 2023

Indice

1 Introduzione **3**

1.1 Riconoscimento Ottico dei Caratteri 4

1.1.1 Componenti fondamentali 4

1.1.2 Reti neurali convoluzionali 5

1.1.3 Il modello ASTER 6

1.2 Qualità delle immagini 9

1.2.1 Raccolta di dati 10

1.2.2 Degradazione artificiale di immagini 10

1.2.3 Degradazione naturale di immagini 11

1.2.4 Metriche di valutazione della qualità 12

1.3 Obiettivo dello studio 14

2 Materiali e metodi **15**

2.1 PyTorch 16

2.2 Dataset e addestramento 16

2.2.1 TextZoom 17

2.2.2 Loss function 17

2.3 TSRN 19

3 Risultati **21**

3.1 ASTER 21
3.2 PSNR e SSIM 23
3.3 Conclusioni 24

4 Discussione e sviluppi futuri **25**

Ringraziamenti **31**

Capitolo 1

Introduzione

Il riconoscimento ottico di caratteri (OCR) e conseguentemente di testi contenuti in immagini e video in formato digitale è ambito di profondo interesse in molti applicativi pratici. Le tecniche utilizzate per l'esecuzione di OCR richiedono i passaggi preliminari di segmentazione e pre-processing del contenuto a monte dell'estrazione di caratteristiche contestuali necessarie al riconoscimento (features), sempre più spesso operata mediante reti neurali profonde.

Nell'ambito dell'analisi di contenuti multimediali di origine amatoriale o contenenti degradazione dovuta alle tecniche di acquisizione, la qualità effettiva determina in modo sostanziale l'accuratezza di questi passaggi e del risultato estratto. Modalità odierne di affrontare il problema concludono l'efficacia dell'aggiunta di un passaggio preliminare: la ricostruzione qualitativa del contenuto mediante aumento di risoluzione e miglioramento della nitidezza originale.

Il compito è affidato a sua volta a modelli basati su reti neurali capaci di predire la struttura dei contenuti assenti.

1.1 Riconoscimento Ottico dei Caratteri

Il riconoscimento ottico dei caratteri è la tecnica più efficace per il recupero testuale di informazioni scritte contenute in immagini. Esso stesso è suddivisibile in molteplici fasi di analisi e permette il riconoscimento di diverse lingue e formati, dalla scrittura a mano a quella pubblicitaria contenuta in insegne, fino al caso apparentemente più semplice del testo digitale di carattere comune.
Ciascuna delle procedure viene convenzionalmente affidata all'elaborazione da parte di moduli basati su reti neurali. Ognuna delle sottoparti deve quindi essere addestrata congruentemente con i suoi fini e mediante quantità di dati, variabili dipendentemente dalla difficoltà del proprio compito.

1.1.1 Componenti fondamentali

L'avvento delle Reti Neurali Profonde ha imposto alle tecniche di riconoscimento testuale il paradigma dei modelli Convoluzionali e Ricorsivi [1], rispettivamente abbreviati con CNN e RNN.
Cuore dei modelli allo stato dell'arte, queste due tecniche ampiamente diffuse non suppliscono specificamente ai problemi di distorsione, degradazione e allineamento che affliggono molte immagini contenenti testo, ma permettono l'acquisizione carattere per carattere o di intere stringhe qualora tali condizioni vengano mitigate da moduli supplementari e specifici per ogni applicativo.
In generale, i modelli di pura estrazione del contenuto necessitano di input che soddisfino certi requisiti per operare con massima accuratezza. Si dimostrano infatti fondamentali le procedure di segmentazione e rettificazione delle immagini da analizzare. La prima operazione permette di delineare e ritagliare, a partire da fotogrammi contenenti soggetti multipli, le aree contenenti testo e di racchiuderle in poligoni detti bounding box, al fine di isolarle. La rettificazione viene poi effettuata

sui soggetti specifici per rimuovere distorsioni causate dalla prospettiva, come mostrato in Fig. 1.1.

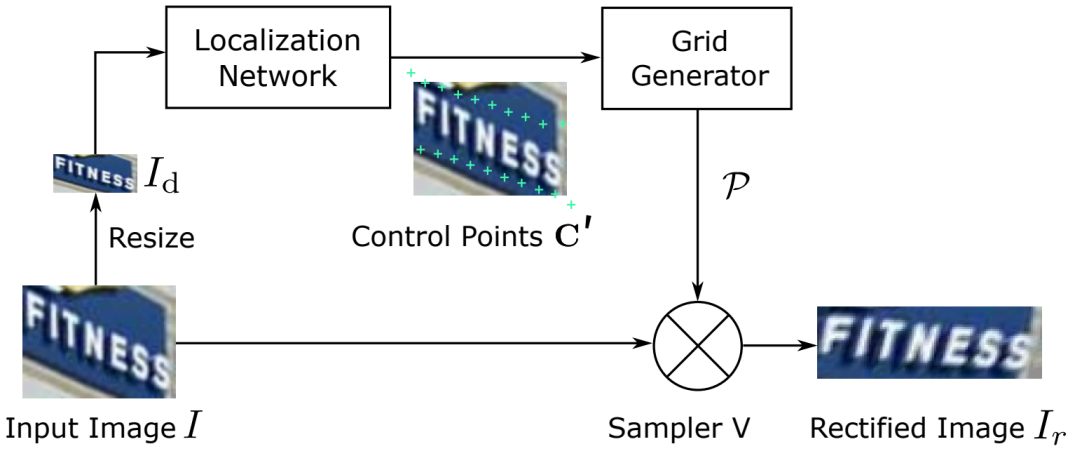

Figura 1.1: *Procedura di rettificazione del modello ASTER* [2]

Il risultato è un'immagine dal formato standard che può essere analizzato dalla restante parte del modello di OCR. Dati ulteriori difetti di sfocatura e scarso dettaglio degli angoli dei caratteri, tecniche ulteriori, dette di Super Resolution, possono essere applicate in questa fase per ottenere un riconoscimento più accurato.

1.1.2 Reti neurali convoluzionali

Il modello di rete neurale assolutamente più diffuso per l'analisi di immagini è quello convoluzionale, abbreviato comunemente con CNN.

Tale paradigma sfrutta l'operazione matematica di convoluzione tra funzioni discrete e permette di estrarre feature da dati la cui rappresentazione nativa, come nel caso di fotogrammi, è matriciale, mantenendo integre le informazioni di carattere spaziale. Si evita in questo modo la vettorializzazione, più tipica per l'input di reti neurali canoniche.

Nel caso più semplice, una rete convoluzionale è composta da strati di ingresso che elaborano l'input e successive parti che ne appiattiscono le uscite rifacendosi alle tecniche classiche degli MLP. La convoluzione di funzioni discrete bidimensionali è definibile come:

$$(f * g)[x, y] = \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} f[x, y] \cdot g[x - i, y - j]$$

Dove nel caso di una CNN: $f[x, y]$, $g[x, y] : \mathbb{R}^2 \longrightarrow \mathbb{R}$ rappresentano rispettivamente l'immagine e il kernel di convoluzione. In particolare per immagini a colori su tre livelli (RGB), altrettante funzioni f dette canali, si modellano singolarmente e se ne calcola la convoluzione con g, un dominio quadrato di dimensioni inferiori ad f. Le funzioni risultanti, che prendono il nome di *feature maps*, vengono poi ulteriormente sottoposte a convoluzione, oppure direttamente vettorializzate per la propagazione a strati di un MLP.

Mediante il calcolo delle loss function stabilite e l'applicazione del noto algoritmo di Backpropagation, è possibile apprendere i valori dei kernel g per risolvere in modo specifico un determinato compito.

Ulteriore vantaggio delle CNN è la possibilità di ridurre le dimensioni degli output di strati convoluzionali mediante la tecnica del *Pooling*, permettendo così la gestione più efficiente di input in alta risoluzione e la maggiore stratificazione a vantaggio del costo computazionale.

1.1.3 Il modello ASTER

Data la natura applicativa degli strumenti di OCR, realizzazioni moderne di carattere onnicomprensivo sono attualmente in fase di sviluppo o già implementate da importanti aziende o centri di ricerca. Tuttavia, il codice sorgente di molte di queste non è reso pubblico. Per questa ragione molti degli studi supplementari in materia

sfruttano come caso studio le migliori prestazioni ottenibili con architetture note e aperte. Questo è il caso di ASTER [2] (*Attentional Scene Text Recognizer with Flexible Rectification*) ma anche degli altrettanto noti MORAN [3] e CRNN [4].
Come suggerito dal nome, ASTER sfrutta una tecnica di rettificazione nota come *Thin-Plate-Spline* (TPS), per elaborare in modo flessibile testi curvi e distorti dalla prospettiva.
Una trasformazione TPS è ottenibile a partire da due insiemi di punti di controllo di uguale numero K, indicati in Fig. 1.2 sulle immagini di partenza, dai quali si ottengono i corrispondenti punti nelle immagini risultanti come margini superiore e inferiore.

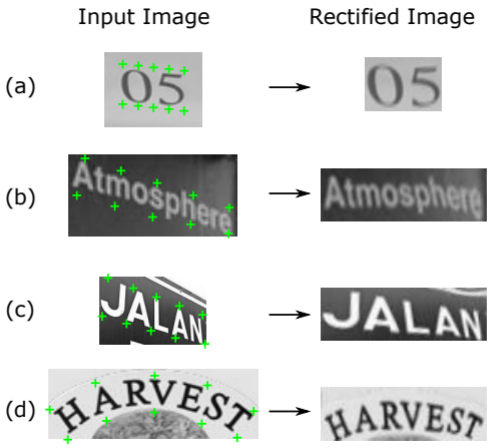

Figura 1.2: *Esempi di rettificazione di immagini operate da ASTER* [2]

Ridotto il problema della rettificazione alla sola predizione di tali punti, questa è affidata ad una rete convoluzionale.
Denotate I e I_r le immagini di input-output e considerati K punti di interesse, si indicano questi con $\mathbf{C}'$ e $\mathbf{C}$:

$$\mathbf{C}' = [c'_1, c'_2, \ldots, c'_K] \in \mathbb{R}^{2\times K} \qquad \mathbf{C} = [c_1, c_2, \ldots, c_K] \in \mathbb{R}^{2\times K}$$

Dove $\mathbf{c}_k = [x_k, y_k]^T$ sono le coordinate del k-esimo punto. La rete convoluzionale costruisce $\mathbf{C}'$ direttamente a partire da I e il suo output viene messo nella forma:

$$\mathbf{C}' \in \mathbb{R}^{2\times K}$$

Tutti i moduli contenuti nella parte di rettificazione sono differenziabili, in modo da poterne aggiustare i parametri in fase di addestramento con l'algoritmo di Backpropagation.

La parte seguente del modello esegue l'OCR sulle immagini rettificate secondo il metodo *Connectionist Temporal Classification* (CTC). Questo provvede alla creazione di una loss function a sua volta differenziabile e insensibile al posizionamento orizzontale dei singoli caratteri. Per trattare le dipendenze tra questi ultimi, ASTER fa affidamento su un ulteriore modello linguistico esterno.

In Fig. 1.3 viene riportata la struttura di base della rete neurale impiegata da ASTER nel riconoscimento, a valle della rettificazione e dell'eventuale pre-processing supplementare.

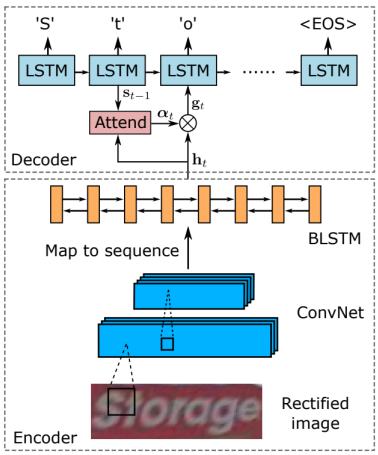

Figura 1.3: *Rete di riconoscimento di ASTER* [2]

1.2 Qualità delle immagini

Nonostante le crescenti capacità dei modelli di OCR odierni, la qualità complessiva delle immagini sottoposte ad analisi rimane di fondamentale importanza per ottenere risultati consistenti in ambiti professionali.

Degradazioni significative dei fotogrammi considerati, causate da condizioni più o meno controllabili e di natura ottica, possono facilmente rendere difficoltoso e in alcuni casi impossibile il recupero dell'informazione. Strumenti moderni, come ad esempio Google Bard, GPT4 o programmi di elaborazione di immagine in generale, integrano già al loro interno tecniche di miglioramento della risoluzione per affinare le prestazioni dei più classici algoritmi di riconoscimento.

Data la natura delle soluzioni si rendono necessarie enormi quantità di potenza di calcolo e dati, appropriatamente etichettati per l'addestramento di reti neurali profonde. Proprio sulla natura di questi ultimi si è concentrata parte della ricerca, la quale si è spostata largamente dalla produzione di immagini degradate artificialmente alla raccolta di esempi reali, scattati volutamente con distorsioni ottiche.

Dalla combinazione di tali considerazioni nasce il problema denominato *STISR in the Wild*, per *Scene Text Image Super-Resolution in the Wild*, esplorato in dettaglio da [5].

1.2.1 Raccolta di dati

L'addestramento di modelli neurali di dimensioni crescenti richiede la raccolta massiccia di dati debitamente classificati e standardizzati. Per ottenere modelli in grado di ricostruire e dunque aumentare la risoluzione di immagini è necessario essere in possesso di due versioni delle stesse:

- Versione degradata o a bassa risoluzione: y
- Versione originale o ad alta risoluzione: $\hat{y}$

La raccolta di queste è di fondamentale importanza e determina la capacità dei modelli stessi di produrre risultati soddisfacenti

1.2.2 Degradazione artificiale di immagini

Data la quantità necessaria, una valida soluzione al problema della raccolta di coppie di immagini bassa risoluzione / alta risoluzione (LR/HR) è la produzione artificiale di effetti di degradazione sulle versioni native. I metodi più comunemente utilizzati sono:

- Aggiunta di rumore Gaussiano Bianco (AWG)
- Passaggio in filtri passa basso per introdurre sfocatura
- BICUBIC down-sampling

Tecniche simili si dimostrano semplici da implementare e poco dispendiose in termini di tempo e potenza di calcolo. Molti degli avanzamenti in materia provengono da studi eseguiti su tali dati e i primi modelli basati su reti convoluzionali si appoggiano pesantemente sulla creazione da zero di immagini successivamente degradate [1] per risparmiare massicciamente sui costi di raccolta.

Sfortunatamente più recenti impieghi della tecnologia di OCR dimostrano però i limiti della degradazione artificiale e ne evidenziano la profonda differenza con esempi di degradazione reale (Fig. 1.4). Metodiche basate sull'addestramento con dati sintetici tendono infatti ad adattarsi a specifici tipi di rumore, divenendo così meno flessibili e inadatte alla ricostruzione di casi naturali.

1.2.3 Degradazione naturale di immagini

Sebbene molto più dispendiosa, la raccolta di immagini naturali è ad oggi la metodica più ragionevole per l'ottenimento di risultati considerevoli. La realisticità degli artefatti di sfocatura e degradazione non è replicabile algoritmicamente e metodi che implementano Super Resolution basata su tali immagini si dimostrano in generale superiori ai precedenti anche su dati di natura artificiale.
A rendere notevolmente più lenta la raccolta è la necessità di molteplici scatti di fotogrammi con diversi obiettivi e a distanze variabili, seguita dall'etichettatura delle coppie con il testo reale. Considerevole è l'apporto introdotto da [5] con la definizione di un nuovo dataset denominato TextZoom, ad oggi il più utilizzato dalla ricerca in questo ambito. Il notevole incremento di difficoltà nella lettura dei testi, in immagini degradate naturalmente, è evidenziato in Fig. 1.4, in cui nei casi peggiori è visibile anche la perdita quasi totale della forma stessa delle singole lettere.

Figura 1.4: *Differenze tra degradazione sintetica e naturale* [5]

1.2.4 Metriche di valutazione della qualità

Con l'obiettivo di ricostruire i testi LR, emerge la necessità di classificarne la qualità relativa rispetto alla versione HR.

Mutuate dai compiti di ricostruzione di immagini in generale, le tecniche che permettono tale valutazione si dividono in due grandi categorie: soggettive e oggettive. Le prime, caratterizzate dalla presenza di un osservatore umano, permettono una valutazione di ampio spettro della qualità percepita ma si dimostrano maggiormente necessarie in ambiti di raffigurazione naturalistica e artistica. Le metodiche di Super Resolution di immagini contenenti testo si affidano più spesso a tecniche oggettive, avvantaggiandosi di maggiore scalabilità e riproducibilità del risultato.

Tra le molteplici sono di più larga diffusione [6]: *Peak Signal-to-Noise Ratio* ($PSNR$) e *Structural Similarity* ($SSIM$).

- **PSNR**

  Data l'immagine originale HR detta I_y, formata da N pixel, e l'immagine ricostruita a partire dalla versione LR detta I_{SR}, il massimo rapporto segnale rumore può essere definito come:

  $$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{L^2}{\text{MSE}} \right)$$

  Considerando l'errore quadratico medio:

  $$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} (I_y - I_{SR})^2$$

  Dove con L si indica il picco del segnale, che nel caso semplificato di immagini in scala di grigi su 8-bit assume il valore $L = 255 = 2^8 - 1$.

  PSNR permette una valutazione della differenza al livello del pixel tra immagine degradata e originale. Data la sua semplicità è considerata la metrica più utile per constatare il risultato della ricostruzione.

- **SSIM**

  Structural Similarity è considerata una metrica percettiva che, a differenza del PSNR, permette di valutare il grado di distorsione di un'immagine. Come suggerisce il nome essa è anche capace di misurare la similarità tra due input. La funzione e le sue sottoparti sono definibili come:

$$\text{SSIM} = \Big(l\left(I_{SR}, I_y\right)^{\alpha} \cdot c\left(I_{SR}, I_y\right)^{\beta} \cdot s\left(I_{SR}, I_y\right)^{\gamma}\Big),$$

$$l\left(I_{SR}, I_y\right) = \left(\frac{2\mu_{I_{SR}}\mu_{I_y} + C_1}{\mu_{I_{SR}}^2 + \mu_{I_y}^2 + C_1}\right),$$

$$c\left(I_{SR}, I_y\right) = \left(\frac{2\sigma_{I_{SR}}\sigma_{I_y} + C_2}{\sigma_{I_{SR}}^2 + \sigma_{I_y}^2 + C_2}\right),$$

$$s\left(I_{SR}, I_y\right) = \left(\frac{\sigma_{I_{SR}I_y} + C_3}{\sigma_{I_{SR}}\sigma_{I_y} + C_3}\right)$$

  Dove σ_I, σ_I^2, σ_{I_xIy} indicano rispettivamente deviazione standard, varianza, covarianza dei pedici e μ_I è la media campionaria per pixel, mentre α, β, γ sono parametri reali utilizzati per attribuire un peso alle parti e C_1 , C_2, C_3 sono valori reali per evitare il caso di denominatore nullo. In particolare $C_1 = (k_1 L)^2$, $C_2 = (k_2 L)^2$, $C_3 = \frac{C_2}{2}$, con $k_1, k_2 \ll 1$

  Si mantengono invece le definizioni di L, I_y e I_{SR} esplicitate in PSNR.

  Nel caso specifico di immagini di differenti risoluzioni, viene utilizzata la versione *Multi Scale* di SSIM (MS-SSIM), che grazie alla visione multivariata permette valutazioni in generale più flessibili.

1.3 Obiettivo dello studio

Le tecniche di Riconoscimento Ottico dei Caratteri sono strumenti in continua evoluzione e applicati in un ampio spettro di ambiti. La raccolta di informazione contenuta in testi direttamente da immagini è diventata di cruciale importanza per l'elaborazione di grandi quantità di dati, ma anche per l'utilizzo quotidiano, tanto da essere ormai una funzionalità distribuita al livello di ogni consumatore, in ogni smartphone e in molti applicativi gratuiti. Moderne soluzioni, sia aziendali che open source, permettono un livello di accuratezza meticoloso su una vasta gamma di testi scritti, a patto però di essere in grado di acquisire fotogrammi di sufficiente qualità, tali da presentare limitate distorsioni e artefatti ottici. Allo stato dell'arte infatti, anche modeste riduzioni di visibilità dei caratteri o stili testuali elaborati possono trarre in errore i software più avanzati, risultando in percentuali di accuratezza limitate per l'uso personale e intrattabili per uso lavorativo. Contesti professionali che necessitino dell'impiego di simili strumenti in ambienti con scarsa luminosità o distanze proibitive, necessitano di metodi di pre-processing dei dati ottenuti, al fine di incrementare la qualità complessiva previo riconoscimento delle informazioni.
Lo scopo che il lavoro di tesi si pone è la comparazione di tecniche esistenti di miglioramento delle immagini e la stima della loro funzionalità in termini di incremento di accuratezza su contenuti degradati da parte dell'OCR. In particolare, si analizzano algoritmi basati su Machine Learning, in grado di aumentare la risoluzione e limitare la dissolvenza dei margini dei caratteri contenuti. L'addestramento dei modelli viene eseguito da zero per replicare risultati pubblicati precedentemente e aggregarne le prestazioni.

Capitolo 2

Materiali e metodi

Le architetture basate su reti neurali esaminate sono state realizzate mediante la nota libreria PyTorch e i test eseguiti si sono svolti in ambiente Linux su una singola scheda grafica per ammortizzarne i costi computazionali, avvalendosi della parallelizzazione offerta dall'infrastruttura CUDA. I risultati sono stati misurati mediante le metriche *PSNR* e *SSIM*, controllandone la differenza con gli obiettivi. L'incremento di accuratezza introdotto è stato invece valutato applicando l'OCR con il modello ASTER precedentemente addestrato.

Nello specifico le architetture esaminate sono state: *VDSR* [7], *SRResNET* [8], *EDSR* [9], *RDN* [10], *LapSRN* [11] e *TSRN* [5]. Ogni avanzamento delle performance è stato seguentemente salvato su file, analizzato e graficato con la libreria Matplotlib. Infine i valori sono stati tabulati per trarne conclusioni quantitative.

2.1 PyTorch

PyTorch è un framework general pourpose per lo sviluppo in ambito Machine Learning in linguaggio Python ed è basato sulla precedente libreria Torch, programmabile in Lua. Viene utilizzato comunemente in applicativi di Computer Vision e Natural Language Processing (NLP) e offre un'interfaccia di alto livello per la gestione di reti neurali di dimensioni e struttura variabili. Oltre alla facilità di utilizzo, permette al programmatore di scrivere software indipendente dall'hardware sottostante e di avvantaggiarsi, con eguale semplicità, di calcolo con schede grafiche (GPU) o processori (CPU), trasformando nel primo caso il codice in forma compatibile con la piattaforma CUDA offerta da Nvidia.

2.2 Dataset e addestramento

L'addestramento dei modelli di Super Resolution si è svolto sulla collezione di dati LR/HR contenuta in TextZoom, introdotta da [5]. Composta da immagini degradate naturalmente, questa si dimostra superiore nell'addestramento alle precedenti istanze prodotte sinteticamente, tra cui le più comuni: *IC03* [12], *IC13* [13], *IC15* [14], *CUTE80* [15], *SVT* [16], *SVTP* [17] e *IIIT5K* [18].

Con l'ipotesi [5] che il modello *TSRN* offrisse le migliori prestazioni, gli iperparametri di training sono stati standardizzati ai requisiti di quest'ultimo. 500 epoche di addestramento sulla debita ripartizione di immagini con tasso di apprendimento fisso al valore 10^{-3} e successiva validazione sulla restante parte.

Le esecuzioni, temporalmente consistenti di circa 24h ciascuna, si sono svolte in tempi separati su una singola GPU Nvidia RTX 4090.

2.2.1 TextZoom

TextZoom è il nome dell'innovativo dataset introdotto dagli stessi ideatori dell'architettura *TSRN* [5]. Figlio dei predecessori RealSR [19] e SRRAW [20], questo si compone di 20121 coppie di immagini LR/HR, suddivise a loro volta in una porzione di training (17367) e una di testing (4373).
Divenuto rilevante come dataset pubblico formato da sole immagini acquisite con degradazione ottica naturale in numero consistente, esso si distingue anche per l'ulteriore categorizzazione dei campioni di test, ripartiti dagli autori nelle 3 categorie:

- **Easy** - 1619 coppie
- **Medium** - 1411 coppie
- **Hard** - 1343 coppie

Così contraddistinte dall'evidente differenza di accuratezza media su di esse ottenibile tramite i modelli ASTER [2], MORAN [3], CRNN [4].

2.2.2 Loss function

Oltre a profondità e struttura modulare, le architetture prese in analisi differiscono anche per l'impiego, in combinazioni variabili, di un totale di 6 *loss function*, denominate L_1, L_2, L_{tv}, L_p, L_c e L_{GP}.
Con le prime tre si contrassegnano rispettivamente le più note: Errore Assoluto Medio (MAE), Errore Quadratico Medio (MSE), Errore alle Variazioni Totali (TVE). Le seguenti vengono introdotte da lavori di ricerca in ambito di Super Resolution:

- L_p **- Perceptual loss** [21]:
  Misura differenze semantiche e percettive di alto livello tra due immagini. Si appoggia a sua volta su una rete convoluzionale profonda pre-addestrata dagli autori di [22].

- L_c **- Charbonnier Loss** [11]:

  Loss function definita e impiegata unicamente dagli autori del modello *LapSRN*. Data x l'immagine LR e dette x_s la versione aumentata di risoluzione a partire da LR, y_s la versione HR e r_s l'immagine residua al livello s, una definizione complessiva di L_c può essere data nella forma:

$$\mathcal{L}(\hat{y}, y, \theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{s=1}^{L}\rho\left(\hat{y}_s^{(i)} - y_s^{(i)}\right) = \frac{1}{N}\sum_{i=1}^{N}\sum_{s=1}^{L}\rho\left(\left(\hat{y}_s^{(i)} - x_s^{(i)}\right) - r_s^{(i)}\right)$$

  Dove θ rappresenta l'insieme dei parametri addestrabili della rete e $\rho(x) = \sqrt{x^2 + \epsilon^2}$ è la funzione di penalizzazione di Charbonnier, variante differenziabile della norma l_2 in [23], N il numero di campioni per batch e L il numero di livelli della piramide definita in [11]. ϵ è empiricamente fissato come 10^{-3}.

- L_{GP} **- Gradient Profile loss** [5]:

  Ispirata dalla precedente *Gradient Profile Prior* (GPP) [24], L_{GP} viene introdotta per migliorare la ricostruzione dei margini dei caratteri.

  Considerando il gradiente spaziale dei valori RGB dei singoli pixel, GPP viene rivisitata per definire:

$$L_{GP} = \mathbb{E}_x||\nabla I_{hr}(x) - \nabla I_{sr}(x)|| \qquad (x \in [x_0, x1])$$

  In cui con I_{hr} e I_{sr} ci si riferisce alle immagini in alta risoluzione e ricostruita con Super Resolution.

In particolare il modello di riferimento TSRN utilizza la combinazione delle sole L_2 e L_{GP}.

2.3 TSRN

Modello più recente e promettente della famiglia presa in considerazione, *Text Super-Resolution Network* (TSRN) [5] risulta in una versione migliorata e ampliata di SRResNet [8]. Visibile in Fig. 2.1, la pipeline mostra i passaggi a cui viene sottoposto l'input LR, diviso nei tre canali RGB, accoppiato con una maschera binaria, ottenuta mediante il valore medio della versione in scala di grigi. Un modulo per l'allineamento viene aggiunto all'architettura originale e i blocchi seguenti sostituiti con *Sequential Residual Blocks* (SRBs), dei quali è visibile il dettaglio. Estratte le feature superficiali in ingresso ed elaborate quelle profonde tramite gli SRBs, una procedura di sovracampionamento estende infine la risoluzione alle dimensioni del corrispondente HR, permettendo così il confronto mediante le loss function L_2, L_{GP}.

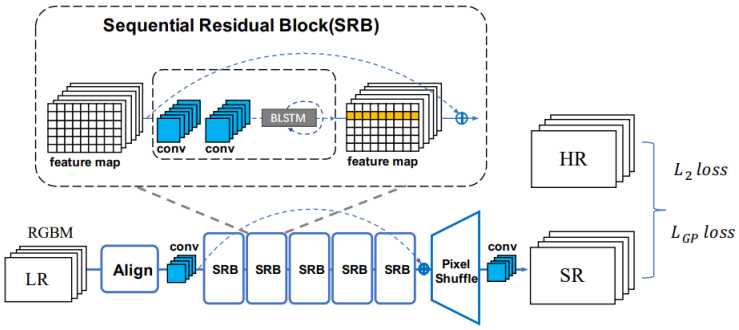

Figura 2.1: *Pipeline del modello TSRN* [5]

Parte dell'innovazione è rappresentata dal miglioramento del blocco *Long-Short Term Memory* (LSTM), nella sua forma bidirezionale (BLSTM). Questo permette la propagazione di errori differenziali e l'inversione delle feature map in *feature sequences*, le quali vengono reintrodotte allo strato convoluzionale. BLSTM viene fornito delle feature verticali e orizzontali sequenzialmente e aggiorna il suo stato interno ricorsivamente.

$$H_{t_1} = \Phi_1 \left(X_{t_1}, H_{t_1 - 1} \right), \qquad t_1 = 1, 2, \ldots, W$$
$$H_{t_2} = \Phi_1 \left(X_{t_2}, H_{t_2 - 1} \right), \qquad t_2 = 1, 2, \ldots, H$$

Qui con H_t si denotano gli hidden layer, X_t è la feature in input, mentre t_1, t_2 indicano la connessione ricorsiva tra le direzioni orizzontale e verticale.

Capitolo 3

Risultati

3.1 ASTER

In fase di test, la metrica tenuta in considerazione come maggiormente rilevante è stata l'accuratezza complessiva di OCR del modello ASTER.

In Tab. 3.1 sono visibili le prestazioni nelle tre categorie — *Easy*, *Medium*, *Hard* —

Metodo	**Accuratezza di ASTER**			
	Easy	**Medium**	**Hard**	**Average**
BICUBIC	61.58%	40.96%	29.71%	44.08%
VDSR [7]	59.23%	40.33%	29.11%	42.89%
SRResNet [8]	61.46%	48.19%	33.51%	47.72%
RDN [10]	66.52%	46.92%	33.73%	49.06%
EDSR [9]	67.33%	48.62%	34.03%	49.99%
LapSRN [11]	48.18%	36.07%	24.94%	36.40%
TSRN [5] (mask off)	67.45%	52.52%	36.56%	52.18%
TSRN [5] (mask on)	**74.12%**	**56.63%**	**39.46%**	**56.74%**

Tabella 3.1: Risultati dei modelli con ASTER OCR

Si riportano anche i valori medi e i risultati ottenuti con l'algoritmo di *BICUBIC interpolation*, non appoggiato da moduli neurali.
In Fig. 3.1 è mostrato l'andamento, in 500 epoche, delle performance tabulate per il modello TSRN [5] nella versione con maschera binaria.

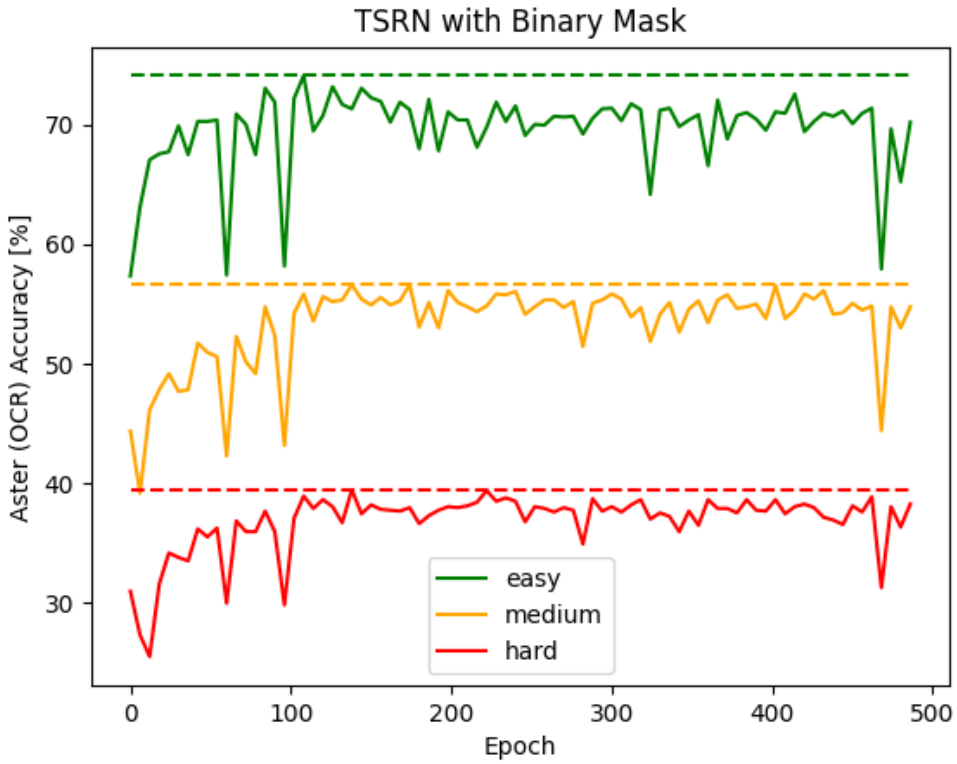

Figura 3.1: *Progresso del training di TSRN sul dataset TextZoom*

Nell'analizzare il grafico è necessario tenere in considerazione che la base di partenza viene conferita dalle già sufficienti capacità di ASTER di riconoscere anche immagini in bassa risoluzione.

3.2 PSNR e SSIM

Seppure di minore rilevanza, durante l'addestramento si è tenuta traccia anche dei migliori valori di similarità LR/HR ottenuti. in Tab. 3.2 si riportano gli indicatori *Peak Signal-to-Noise Ratio* (PSNR) e *Structural Similarity* (SSIM).

Metodo	**PSNR**			**SSIM**		
	Easy	**Medium**	**Hard**	**Easy**	**Medium**	**Hard**
BICUBIC	22.33	18.96	19.42	0.79	0.63	0.66
VDSR [7]	21.98	19.06	19.3	0.77	0.62	0.65
SRResNet [8]	21.01	18.28	18.60	0.79	0.62	0.66
RDN [10]	23.48	19.12	**20.02**	0.85	0.65	0.71
EDSR [9]	**23.60**	19.08	20.01	0.85	0.65	0.72
LapSRN [11]	14.06	14.11	13.43	0.69	0.56	0.60
TSRN [5] (mask off)	22.09	18.97	19.38	0.82	0.63	0.69
TSRN [5] (mask on)	23.48	**19.17**	20.01	**0.87**	**0.67**	**0.74**

Tabella 3.2: *Risultati dei modelli con ASTER OCR*

3.3 Conclusioni

Con valori di accuratezza media del 56.74% di ASTER [2], L'architettura *TSRN* [5], grazie anche all'integrazione della maschera binaria, si dimostra modestamente superiore rispetto a tutte le altre prese in considerazione.

Nonostante la mancanza di addestramenti con variazioni significative degli iperparametri, i risultati si dimostrano concordi con le evidenze già presenti nella precedente letteratura in materia. Da tenere in considerazione, però, è l'esistenza di ulteriori migliorie più recenti che apporterebbero, secondo gli autori, incrementi significativi di performance anche a partire dalla base di [5]. La mancanza di un confronto diretto tra queste e i modelli proposti deriva dall'attuale assenza di codice pubblico, il quale però potrebbe rivelarsi disponibile nel prossimo futuro. Tra i metodi più promettenti si citano: *STT* [25], *PCAN* [26], *TG* [27], *TATT* [28], *C3-STISR* [29], *TATSR* [30]

Capitolo 4

Discussione e sviluppi futuri

Nella sua formulazione più generica, il problema dell'implementazione di metodi per Riconoscimento Ottico dei Caratteri, è lontano dalla sua soluzione definitiva. Risultati significativi sono ottenibili ad oggi solamente in presenza di materiale strutturato e di sufficiente qualità. La presenza nei contenuti di sofisticazioni stilistiche, distorsioni, degradazioni o sfocature può rendere difficoltoso il compito per ogni modello, neurale o meno, tra i più avanzati. Attualmente, le prestazioni migliori sono probabilmente raggiunte da modelli generalisti di dimensioni aziendali e di portata non replicabile su hardware di livello consumer. Strumenti odierni come i *Large Language Model* (LLM) offerti da OpenAI e Google, hanno assunto in tempi recentissimi capacità multimodali, tali da elaborare enormi quantità di immagini e testi ed estrarne contenuti di notevole difficoltà. Ciò nonostante molte delle immagini LR contenute ad esempio in TextZoom sono ancora perfettamente capaci di sfidare anche le più massicce architetture. Simili conclusioni lasciano poco respiro alla speranza di piccoli team di ricerca di ottimizzare tecniche di OCR a livelli professionali. La strada rimane percorribile invece per soluzioni specifiche che necessitino di contenute risorse di calcolo e che soprattutto forniscano una standardizzazione di qualità e accuratezza richieste.

In termini generali, allo stato dell'arte, sviluppi futuri delle tecnologie di OCR puntano nella direzione di architetture complesse, capaci se affiancate da sufficienti risorse, di generalizzare le loro abilità mediante informazioni contestuali. Tra i vari è impossibile non citare il paradigma dei *Transformer*, radice strutturale dei più promettenti LLM.

Nonostante i grandi passi in avanti degli ultimi anni, lo sviluppo in ambito rimane un problema aperto, sulla quale si concentrano numerosi gli sforzi di ricercatori nella sfera del Machine Learning e in generale dell'intelligenza artificiale.

Bibliografia

[1] Max Jaderberg, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Synthetic data and artificial neural networks for natural scene text recognition, 2014.

[2] Baoguang Shi, Mingkun Yang, Xinggang Wang, Pengyuan Lyu, Cong Yao, and Xiang Bai. Aster: An attentional scene text recognizer with flexible rectification, 06 2018.

[3] Canjie Luo, Lianwen Jin, and Zenghui Sun. A multi-object rectified attention network for scene text recognition, 2019.

[4] Baoguang Shi, Xiang Bai, and Cong Yao. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition, 07 2015.

[5] Wenjia Wang, Enze Xie, Xuebo Liu, Wenhai Wang, Ding Liang, Chunhua Shen, and Xiang Bai. Scene text image super-resolution in the wild, 2020.

[6] Xuan Wang, Jinglei Yi, Jian Guo, Yongchao Song, Jun Lyu, Jindong Xu, Weiqing Yan, Jindong Zhao, Qing Cai, and Haigen Min. A review of image super-resolution approaches based on deep learning and applications in remote sensing, 2022.

[7] Jiwon Kim, Jung Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks, 11 2015.

[8] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. pages 105–114, 07 2017.

[9] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution, 2017.

[10] Yulun Zhang, Yapeng Tian, Yu Kong, and Bineng Zhong. Residual dense network for image super-resolution. pages 2472–2481, 06 2018.

[11] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution, 2017.

[12] Simon M. Lucas, Alex Panaretos, Luis Sosa, Anthony Tang, Shirley Wong, Robert Young, Kazuki Ashida, Hiroki Nagai, Masayuki Okamoto, Hiroaki Yamamoto, Hidetoshi Miyao, JunMin Zhu, WuWen Ou, Christian Wolf, Jean-Michel Jolion, Leon Todoran, Marcel Worring, and Xiaofan Lin. Icdar 2003 robust reading competitions: entries, results, and future directions., 2005.

[13] Dimosthenis Karatzas, Faisal Shafait, Seiichi Uchida, Masakazu Iwamura, Lluis Gomez, Sergi Mestre, Juan mas romeu, David Mota, Jon Almazan, and Lluís-Pere Heras. Icdar 2013 robust reading competition. pages 1484–1493, 08 2013.

[14] Dimosthenis Karatzas, Lluis Gomez-Bigorda, Anguelos Nicolaou, Suman Ghosh, Andrew Bagdanov, Masakazu Iwamura, Jiri Matas, Lukas Neumann, Vijay Ramaseshan Chandrasekhar, Shijian Lu, Faisal Shafait, Seiichi Uchida,

and Ernest Valveny. Icdar 2015 competition on robust reading. In *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*, pages 1156–1160, 2015.

[15] Anhar Risnumawan, Palaiahankote Shivakumara, Chee Seng Chan, and Chew Lim Tan. A robust arbitrary text detection system for natural scene images, 2014.

[16] Kai Wang, Boris Babenko, and Serge Belongie. End-to-end scene text recognition. pages 1457–1464, 11 2011.

[17] Trung Quy Phan, Palaiahnakote Shivakumara, Shangxuan Tian, and Chew Lim Tan. Recognizing text with perspective distortion in natural scenes. In *2013 IEEE International Conference on Computer Vision*, pages 569–576, 2013.

[18] Anand Mishra, Karteek Alahari, and C. Jawahar. Scene text recognition using higher order language priors, 09 2012.

[19] Jianrui Cai, Hui Zeng, Hongwei Yong, Zisheng Cao, and Lei Zhang. Toward real-world single image super-resolution: A new benchmark and a new model, 2019.

[20] Xuaner Cecilia Zhang, Qifeng Chen, Ren Ng, and Vladlen Koltun. Zoom to learn, learn to zoom, 2019.

[21] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016.

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[23] Andres Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: Combining local and global optic flow methods, 02 2005.

[24] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Gradient profile prior and its applications in image super-resolution and enhancement, 11 2010.

[25] Jingye Chen, Bin Li, and Xiangyang Xue. Scene text telescope: Text-focused scene image super-resolution. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12021–12030, 2021.

[26] Cairong Zhao, Shuyang Feng, Brian Nlong Zhao, Zhijun Ding, Jun Wu, Fumin Shen, and Heng Tao Shen. Scene text image super-resolution via parallelly contextual attention network. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 2908–2917, New York, NY, USA, 2021. Association for Computing Machinery.

[27] Jingye Chen, Haiyang Yu, Jianqi Ma, Bin Li, and Xiangyang Xue. Text gestalt: Stroke-aware scene text image super-resolution, 2021.

[28] Jianqi Ma, Zhetong Liang, and Lei Zhang. A text attention network for spatial deformation robust scene text image super-resolution, 2022.

[29] Minyi Zhao, Miao Wang, Fan Bai, Bingjia Li, Jie Wang, and Shuigeng Zhou. C3-stisr: Scene text image super-resolution with triple clues, 2022.

[30] Rui Qin, Bin Wang, and Yu-Wing Tai. Scene text image super-resolution via content perceptual loss and criss-cross transformer blocks, 2022.

Ringraziamenti

Al termine di questo elaborato, desidero esprimere la mia profonda gratitudine al Prof. Marco Maggini per la disponibilità e per l'ispirazione nella scelta del progetto e dei miei futuri studi in materia. Un sentito ringraziamento va ai Dott. Simone Bonechi, Dott. Paolo Andreini per l'assistenza tecnica durante le fasi preliminari e specialmente al Dott. Marco Tanfoni, per la meticolosa supervisione e l'indirizzamento del lavoro. Ringrazio, inoltre, ogni membro del laboratorio dell'Università di Siena per il supporto ricevuto e per gli efficienti mezzi forniti.

Un enorme ringraziamento va ai miei genitori che mi hanno da sempre sostenuto e motivato. Mai mi sono sentito in difetto per i fallimenti e sempre ho sentito la vostra gioia per i miei successi. Vi ringrazio per aver creduto in me e avermi spronato a continuare sulla mia strada anche nei momenti peggiori. Grazie a mio fratello Alessandro con cui ho condiviso gli istanti più felici dall'infanzia fino ad oggi, ti auguro il meglio per il futuro con tutto l'affetto che ci lega, ti ringrazio sapendo per certo che potrò sempre contare su di te. Grazie a mio nonno Vasco e mia nonna Grazia per la vostra stima e la vostra curiosità, per il costante interesse nell'andamento del mio percorso. Sono grato a tutta la mia famiglia per essermi stata vicina da sempre. Non sarei nulla di quello che sono oggi senza la fortuna di avervi accanto.

Ringrazio tutte le amicizie più strette per le esperienze passate insieme, per tutte le risate e per ogni singolo racconto o discussione. È anche grazie alla vostra vicinanza se oggi posso dire di aver raggiunto questo mio obiettivo. Grazie Riccardo, Giovanni, Niccolò, Duccio, Federico, Nicola, Filippo, Pietro, Emilio, Edoardo, Riccardo, Ludovico; mi avete sempre stimato e sostenuto. Grazie a tutti per aver rallegrato le mie giornate più pesanti. Uno speciale ringraziamento a Lorenzo, compagno in questa avventura dall'inizio alla fine, il tempo insieme mi ha insegnato quanto conti la determinazione, hai alleggerito con tenacia e umorismo ognuna delle estenuanti sedute di studio, esorcizzando il mio pessimismo. Grazie a tutti i colleghi e amici in Università con cui ho condiviso le ansie e i successi. Grazie Luca, Giulio, Kevin, Fabrizio, Jacopo, Andrea, Tommaso, Olga, Ginevra. I miei auguri per la carriera migliore possibile e per tante altre soddisfazioni.

Infine ringrazio profondamente la persona che più mi è stata accanto durante l'ultima parte di questo percorso, che mi ha supportato, ma soprattutto sopportato, in tutto e che mi regala quotidiane gioie. Grazie Mimosa, grazie dell'amore che ci regaliamo continuamente e della grande stima che nutri per me.