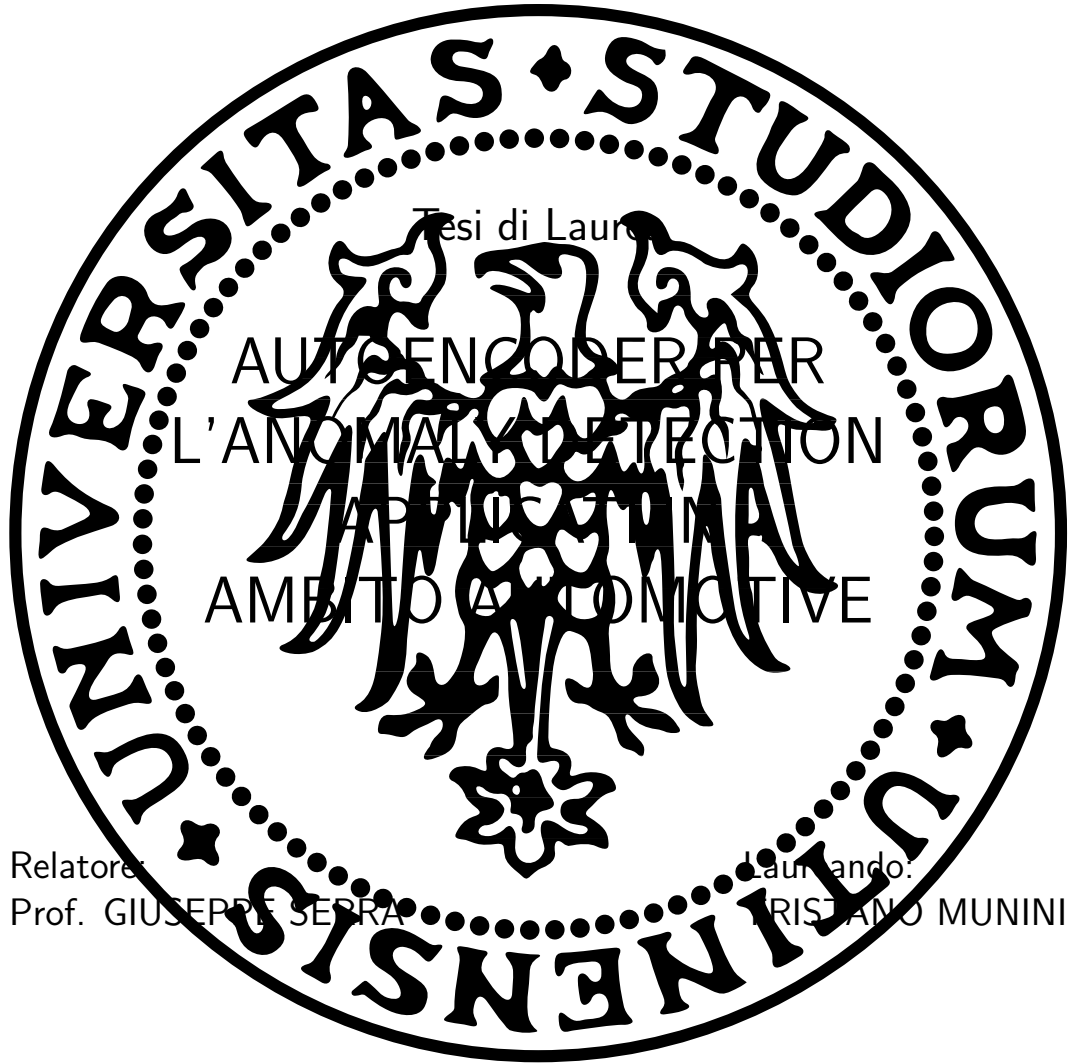


UNIVERSITÀ DEGLI STUDI DI UDINE

Dipartimento di Scienze Matematiche, Informatiche e Fisiche

Corso di Laurea Triennale in Informatica



Relatore:
Prof. GIUSEPPE SERRA

Laurando:
CRISTIANO MUNINI

ANNO ACCADEMICO 2018-2019

Ai miei genitori
per non avermi tagliato i viveri

Abstract

TODO

Introduzione

In campo automotive, soprattutto negli ultimi anni, si è visto un crescente interesse nei confronti di sistemi di intelligenza artificiale che permettano di supervisionare la qualità dei pezzi prodotti. Assicurare la qualità è un requisito critico perché, in generale, la qualità influenza l'intera vita di un prodotto. Le applicazioni di algoritmi di Machine Learning prima, e di sistemi di Deep Learning poi, si sono dimostrate efficaci, flessibili e resilienti, portando numerosi vantaggi non solo nel campo del controllo automatico ma anche in quello del supporto agli operatori umani, ad esempio. I riferimenti alle applicazioni ben riuscite di sistemi intelligenti crescono di mese in mese ed offrono un ottimo mercato. I riscontri in un mercato

TODO

Indice

Abstract	v
Introduzione	vii
Elenco delle figure	xi
1 Il Problema	1
1.1 Le Carcasce per Motori Elettrici	1
1.2 Il Processo di Produzione	2
1.3 Gli Obiettivi da Raggiungere	3
2 Il Dataset	5
2.1 Problematiche Principali	5
2.1.1 Dataset Piccolo e Sbilanciato	5
2.1.2 Differenze tra Immagini	7
2.2 Importanza Preprocessing	9
2.2.1 Centramento mediante Hough Circles	9
2.2.2 Passaggio da RGB a GrayScale	9
2.2.3 Equalizzazione	9
2.2.4 Bilateral Filtering	9
2.2.5 Masking	9
2.2.6 Crop e Resizing	9
2.3 Data Augmentation	9
2.3.1 Rotazione	9
2.3.2 Generazione Scarti Sintetici	9
3 Gli Autoencoder	11
3.1 La Struttura di un Autoencoder	11
3.1.1 Variazioni dell'architettura	12
3.1.2 Applicazioni principali	12
3.2 Convoluzioni e Convoluzioni Trasposte	12

3.3 Spazi Latenti	12
4 Valutazione Sperimentale	13
5 Risultati Ottenuti	15
6 Metodi Alternativi	17
7 Ulteriori Risultati	19
8 Conclusioni	21
A Come si fanno le appendici	23
B Esempi di Citazioni Bibliografiche	25
C Ambiente GNU/Linux (ad esempio Ubuntu)	27
Bibliografia	29

Elenco delle figure

1.1	Visioni laterale e superiore di una carcassa	1
-----	--	---

Capitolo 1

Il Problema

In questo documento si affronta il problema di rilevare la presenza di colla all'interno di carcasse per motori elettrici, per mezzo di fotografie digitali. Dopo alcune considerazioni sulla forma e sul fine ultimo di tali pezzi seguirà una descrizione sommaria del processo industriale e delle macchine che manipolano e depositano la colla all'interno dei pezzi, nonché del sistema di acquisizione immagini. Si conclude la sezione con una formalizzazione del problema da risolvere e le metriche con cui valutare la soluzione proposta.

1.1 Le Carcasse per Motori Elettrici



Figura 1.1: Visioni laterale e superiore di una carcassa

Osservando le fotografie in figura, possono essere definite le caratteristiche principali delle carcasse:

- il pezzo ha una struttura cilindrica cava;

- il fondo presenta tre gradini;
- sono presenti due balze (nella foto laterale se ne vede soltanto una), posizionate, una di fronte all'altra, prima del primo gradino;
- i supporti alla bocca della carcassa presentano due fori;

Il pezzo è stato pensato per avvolgere e proteggere motori elettrici. Nello specifico i pezzi in foto, sui quali è stato svolto lo studio, sono carcasse per motori elettrici per tergicristalli. I due fori aiuteranno a fissare con delle viti il pezzo su dei supporti in plastica. Il motore alloggerà interamente nella cavità, nella quale verrà anche depositata della colla. Questa verrà poi cotta in modo che il motore non possa vibrare all'interno della carcassa, evitando che eventuali urti possano danneggiarlo. Dalla foto si può notare che la colla è distribuita in modo da formare un anello ad una altezza di circa $4cm$ dal fondo della carcassa. La colla è stata depositata correttamente se:

1. l'anello non presenta né sbavature né discontinuità;
2. sul fondo non c'è presenza di colla.

In questo documento, come si vedrà nella sezione sul dataset, ci concentreremo esclusivamente sul secondo punto. La presenza di colla sul fondo della carcassa causerà il malfunzionamento del motorino dopo un limitato tempo d'utilizzo, molto inferiore al tempo di vita atteso. Per questo motivo è fondamentale che la colla venga depositata correttamente.

1.2 Il Processo di Produzione

Chiarire le modalità con cui le carcasse vengono manipolate, la colla viene depositata e le foto vengono acquisite risulta fondamentale. Senza queste informazioni mancherebbe la base sulla quale costruire ipotesi e considerazioni riguardo le immagini del dataset. Analizzando le condizioni in cui le foto vengono scattate, si definiscono i vincoli ed i confini entro i quali le soluzioni proposte possono considerarsi verosimili ed applicabili al mondo reale.

Le carcasse, già presenti in grandi quantità in magazzino, raggiungono il macchinario e vengono caricate, con la concavità rivolta verso l'alto, su un disco rotante. Ad ogni ciclo macchina la colla, tramite due ugelli, viene depositata simultaneamente su due carcasse distinte. Al contempo due sonde dotate di luce scendono nelle due carcasse su cui la colla era stata depositata il ciclo precedente, fino ad una distanza di circa $3cm$ dal fondo. Le carcasse ritenute conformi procedono lungo un rullo trasportatore, mentre quelle che non idonee vengono scartate.

Vanno precisati vari aspetti. Le carcasse, nonostante siano tutte dello stesso tipo, possono differire per quanto riguarda colore, graffi superficiali, sporco, incrostazioni oppure macchie. Inoltre non vengono orientate tutte allo stesso modo rispetto all'asse verticale: questo ha delle ripercussioni dirette sulle foto raccolte, infatti le due balze non si presenteranno in posizioni fisse.

Il sistema assicura che le foto vengano scattate sempre alla stessa profondità e che la sonda sia centrata rispetto al pezzo (considerando come centro il centro della cavità cilindrica). La distanza fissa è condizione sufficiente per garantire la messa a fuoco di ognuno dei tre gradini sul fondo. Purtroppo non sono stati specificati dei vincoli riguardo l'illuminazione.

Si fa notare che il processo appena descritto viene eseguito da almeno due macchinari distinti, ovviamente questo aggiunge un ulteriore grado di sfida: non si può supporre che i macchinari siano sempre calibrati esattamente allo stesso modo.

In conclusione le foto che saranno da analizzare vengono raccolte da un totale di quattro fotocamere distinte, delle quali si assicura

- con un grado di precisione soddisfacente la distanza dal fondo;
- con un grado di precisione accettabile la centratura delle immagini.

1.3 Gli Obiettivi da Raggiungere

Di seguito sono riportati alcuni dati numerici riguardo i processi appena descritti.

La colla viene depositata su circa 5000 pezzi al giorno, la probabilità che gocce di colla cadano sul fondo delle carcasse è estremamente bassa. Purtroppo non esistono dati numerici esatti ma si stima che il macchinario abbia un *fault rate* di una carcassa al mese o poco più. Questi dati possono essere trasformati in probabilità approssimative osservando che:

$$\begin{array}{ll} \text{colla depositata al mese:} & 5000 * 31 = 155000 \\ \text{colla mal depositata al mese:} & 2 \end{array}$$

Quindi la probabilità che il macchinario sbagli è pari allo 0.00001%. Il sistema di intelligenza artificiale deve riconoscere i pezzi non conformi ma soprattutto, tenendo conto della probabilità di cui sopra, deve generare un numero bassissimo di falsi positivi. Ricordano che per falsi positivi (riferiti anche come FP o False Positive) si intendono tutte le carcasse che il sistema considera non conformi ma che in realtà non presentano difetti. Una AI troppo rigida,

che, quando indecisa, propende per scartare il pezzo, inciderebbe negativamente sulla produzione. Si rischierebbe, infatti di creare un enorme danno economico, andando a scartare molte più carcasse del necessario.

Il nostro obbiettivo è quindi quello di generare un numero di falsi positivi che sia inferiore al 2%, cercando di riconoscere più carcasse non conformi possibili.

Capitolo 2

Il Dataset

Innanzitutto si specifica che con la parola Scarto si indica l'immagine di una carcassa che presenta colla sul fondo, pezzo che, quindi, dovrà essere scartato. Invece con la parola Conforme si indica l'immagine di una carcassa nella quale la colla è stata depositata correttamente, quindi con fondo pulito.

TODO recap capitolo In questo capitolo verrà descritto il dataset e quali.

2.1 Problematiche Principali

2.1.1 Dataset Piccolo e Sbilanciato

La prima difficoltà insorge ancora prima di ispezionare le immagini del dataset. Infatti il dataset non solo comprende solamente 1719 immagini ma è anche fortemente sbilanciato:

- 1719 immagini sono Conformi;
- 30 immagini sono Scarti.

A questo punto è corretto chiedersi se le quasi duemila immagini del dataset siano sufficienti ai nostri scopi. Nel campo del *Machine Learning*, ed ancora di più in quello del *Deep Learning*, non è raro che il numero di elementi in un dataset sia dell'ordine delle decine di migliaia se non di quello delle centinaia di migliaia. Basti pensare ai dataset più famosi ed usati:

- MNIST è un dataset molto famoso contenente 70000 cifre disegnate a mano appartenenti a 10 classi in totale. È alla stesso tempo sia il punto di partenza dei principianti, perché di facile manipolazione, sia il campo di prova degli esperti, sul quale vengono allenati nuovi modelli prima di passare a compiti più complessi;

- CIFAR-10 contiene TODO descrivere
- ImageNet contiene TODO descrivere

Ciascuno di questi dataset è stato etichettato¹ a mano. Ciò significa che ad ogni immagine è stata assegnata, a mano, una classe di appartenenza. Prendendo in esempio MNIST, se un'immagine raffigura la cifra 7 allora sarà etichettata con il label *seven* ed apparterrà alla classe di immagini in cui compare la cifra 7. Allo stesso modo le immagini di ImageNet hanno etichette come *dog*, *cat*, *bird*, *car*, *bike*, ecc... Il nostro dataset, come già illustrato, contiene due classi: Conforme e Scarto.

Sembrerebbe che possedere 2000 immagini appena renda impossibile applicare algoritmi di *Machine Learning*. In realtà osservando i Conformi, in figura TODO sono stati riportati alcuni esemplari significativi, ci si accorge che i pezzi sono molto simili. Le differenze principali tra un'immagine e l'altra riguardano la posizione delle balze, la luminosità e le imperfezioni superficiali (graffi, macchie, ...), ciascuna di queste differenze verrà analizzata nel dettaglio tra poco. Quindi, come ci si poteva aspettare essendo pezzi creati meccanicamente, la loro distribuzione è nell'intorno del pezzo progettato. TODO dire meglio. Concludiamo che il numero di Conformi a nostra disposizione è sufficiente

Purtroppo non possiamo dire lo stesso per gli Scarti. Se già la statistica ci lascia sospettare che 30 esemplari non possono ritenersi significativi, allora questo sospetto diventa certezza quando si analizzano le caratteristiche della colla nelle immagini Scarto. Come si vede in figura TODO la colla può presentarsi in forma di gocce più o meno circolari oppure come sbaffi di grossezza e lunghezza variabili. Anche la quantità di superficie coperta dalla colla può variare notevolmente, passando da aree ridotte e localizzate ad aree estese e di conformazioni singolari. Infine notiamo che la posizione del rimasuglio di colla all'interno della carcassa non è in relazione con la posizione delle balze e che la presenza dei gradini sul fondo non la obbliga in alcun modo a scivolare fino al centro.

Per analizzare meglio le modalità con cui potrebbe essere generato uno Scarto si supponga che il macchinario abbia commesso un errore: dall'ugello è uscita una certa quantità di colla in esubero. A seconda della posizione dell'ugello rispetto alla carcassa si può immaginare che la colla raggiunga il fondo in vari modi, proviamo ora ad esplorarne due:

- nel primo caso si immagina che il braccio abbia già depositato l'anello di colla e che si stia allontanando dalla carcassa. La colla in esubero

¹in inglese *labeled* da *label*, etichetta

cadrebbe sotto forma di gocce fino a raggiungere il fondo del pezzo. Questo potrebbe essere il caso per la figura TODO ref immagine con colla a goccia;

- nel secondo caso si immagina che la colla in eccesso faccia parte dell'anello appena depositato e che, a causa delle vibrazioni o di altri fattori simili, colli raggiungendo il fondo della carcassa. Questo potrebbe essere il caso per la figura TODO ref immagine con colla "sbuffata" dal bordo.

Concludiamo che gli esemplari forniti per la classe Scarto descrivono soltanto in modo parziale la distribuzione della classe (TODO dire meglio) e che quindi non possono essere utilizzati per allenare un modello veramente generale. Infatti se, per esempio, venissero usati per il training di una rete convolutiva, di cui poi verrà illustrata brevemente la struttura, si rischierebbe di creare un modello con forte *overfit* rispetto a quelle specifiche macchie di colla fornite. Con il termine *overfit* si intende TODO.

Prima di proseguire con le prossime problematiche dobbiamo spendere alcune parole per commentare la colorazione delle immagini rispetto ai veri colori delle carcasse e della colla. In figura 1.1 a pagina 1 abbiamo visto che la superficie del pezzo è di colore grigio ma nella foto risulta di colore verdastro. Allo stesso modo anche la colla, in realtà di colore bianco sporco, nella fotografia assume tonalità verdognole. Per certi compiti possedere immagini in falsi colori può risultare problematico ma fortunatamente non è questo il caso: l'importante è che venga mantenuta l'informazione che ci permette di distinguere la colla dalla superficie della carcassa. Come vedremo poi le immagini verranno trasformate in scala di grigi quindi, nonostante sarebbe stato preferibile avere immagini a colori reali, i falsi colori non sono da considerarsi problematici.

2.1.2 Differenze tra Immagini

Ora che abbiamo una visione d'insieme sul dataset possiamo concentrare la nostra attenzione sulle proprietà principali delle immagini. Innanzitutto ogni immagine ha una risoluzione di 896x896 pixel, dimensione che ci permette di esplorare varie possibilità. Ad esempio si può pensare di ridurre l'immagine ad una dimensione tale da: occupare meno spazio in memoria, quindi in RAM durante il training, ed allo stesso tempo di mantenere un livello di dettaglio sufficiente ai nostri scopi, risultando quindi in un boost in velocità di training. TODO dire meglio. Oppure di suddividere l'immagine in quadranti da analizzare singolarmente così da mantenere la qualità dell'immagine originale ma senza dover creare una rete che accetti immagini troppo grandi. Infatti una rete che accetta immagini di grandi dimensioni, solitamente, avrà un numero

di parametri maggiore di una che accetta immagini piccole. Questo porta non solo ad occupare più spazio in memoria ma significa anche che la rete impiegherà più tempo sia in fase di train (più parametri da aggiustare) sia in fase di predizione (più conti da fare). TODO sistemare. Per avere un termine di paragone basti pensare che le immagini di MNIST sono 64x64 pixel mentre quelle di ImageNet di $TODO \times TODO$ px.

Osservando nuovamente Figura TODO ci si accorge che le immagini hanno varie proprietà, verranno ora elencate e commentate in ordine crescente di fastidiosità (TODO fix).

- Ogni immagine presenta tre circonferenze concentriche, con centro il centro del pezzo. Ciascuna circonferenza è definita da una transizione da una zona più scura ad una più chiara. Sappiamo che le zone più scure corrispondono alle pareti verticali del pezzo mentre le zone chiare ai tre gradini sul fondo. Questa proprietà non è problematica, anzi potrà essere sfruttata a nostro vantaggio.
- La distanza dal fondo è sempre quella (quindi le dimensioni relative degli anelli).
- Le due balze sulla parete verticale sono ben visibili e possono presentarsi, sempre una di fronte all'altra, in ogni posizione lungo una circonferenza di raggio pari al raggio della cavità cilindrica. Possono essere considerate un problema in quanto rappresentano informazione superflua e variabile. Ricordiamo che la posizione delle balze non ha alcuna correlazione con la presenza della colla, tanto meno con la sua posizione.
- La superficie del pezzo varia ma bene o male è sempre quello. Nota sulle macchie scure.
- centratura (varia ma sappiamo che non può essere completamente fuori)
- riflessività del centro fastidiosa
- luminosità (ci possiamo basare sul più e sul meno luminoso per avere più o meno il range) fastidiosa

Ora possiamo elencare le proprietà esclusive degli Scarti, sempre in ordine di fastidiosità:

- la colla ha texture differente
- la colla è localizzata
- Scarti sono asimmetrici

2.2 Importanza Preprocessing

2.2.1 Centramento mediante Hough Circles

2.2.2 Passaggio da RGB a GrayScale

2.2.3 Equalizzazione

2.2.4 Bilateral Filtering

2.2.5 Masking

2.2.6 Crop e Resizing

2.3 Data Augmentation

Due metodi principali: - rotazione - generazione degli scarti sintetici

2.3.1 Rotazione

Rotazione è metodo standard e largamente usato e bla bla

2.3.2 Generazione Scarti Sintetici

Scarti Sintetici I ritagli non sono stati semplicemente incollati sui conformi: la luminosità della colla è stata modificata per avvicinarsi a quella del pezzo conforme; dopo aver aggiunto la colla è stato praticato uno smooth lungo il contorno, per evitare che ci fosse una transizione netta fra sfondo ed inizio bordo della colla.

Capitolo 3

Gli Autoencoder

3.1 La Struttura di un Autoencoder

TODO quote from wikipedia

Un Autoencoder (AE) è un particolare tipo di rete neurale il cui scopo è codificare efficacemente l'informazione fornita in input con modalità unsupervised. Gli Autoencoder sono stati largamente utilizzati come tecniche per la dimensionality reduction, dimostrando capacità paragonabili a quelle di algoritmi come PCA o t-sne. Un Autoencoder è formato da due componenti principali:

- l'encoder, ottimizzato per generare una versione compressa del dato;
- il decoder, ottimizzato per ricostruire l'informazione originale a partire dalla versione compressa.

Durante l'allenamento (training) la rete cerca di fornire un output il più simile possibile all'input ricevuto. Potrebbe sembrare che l'obiettivo dell'Autoencoder sia simulare la funzione identità, cioè quella funzione che fornisce in output esattamente l'input ricevuto. In realtà all'interno della rete accade molto di più. Prendiamo in esempio il più semplice degli AE: due layer fully-connected messi in sequenza. Sappiamo che il numero di feature ¹ in ingresso nel primo layer deve combaciare con il numero di feature in uscita, altrimenti non potremmo confrontare l'input con la ricostruzione generata dalla rete. Se si immaginano le feature in ingresso come dei punti, vettori, in un spazio n-dimensionale, con n pari al numero delle feature fornite, il compito dell'Autoencoder è generare dei punti che siano molto vicini ai punti osservati. In altre parole si vuole che l'Autoencoder catturi la distribuzione probabilistica dei vettori forniti. Stabilito, quindi, che la rete riceve e restituisce vettori di

¹TODO spiegare cosa si intende per feature

dimensionalità n si può definire la dimensionalità dello spazio latente, cioè di quello spazio di dimensione m , con $m < n$, in cui tutti i vettori in input vengono mappati durante l'encoding. Viene chiamato spazio latente perché non è conosciuto a priori. Infatti sarà proprio la rete stessa, durante il training, a trovarlo, selezionandolo tra gli infiniti spazi m -dimensionali. La dimensione dello spazio latente deve essere sufficientemente grande da permettere di mantenere le informazioni che caratterizzano l'input, ed allo stesso tempo sufficientemente piccolo così da rimuovere eventuale rumore o dati superflui. Poco fa abbiamo definito due layer fully-connected, specificando soltanto le dimensioni in input del primo e le dimensioni in output del secondo. Ora sappiamo che le feature in uscita dal primo layer dovranno essere accettate, in entrata, dal secondo. Sappiamo anche che la dimensione della feature, in quanto vettore dello spazio latente, dovrà essere pari ad m . La rete appena definita e raffigurata in con la tipica forma a clessidra. La parte più stretta di un Autoencoder è detta bottleneck (in italiano: collo di bottiglia) e corrisponde con la compressione massima del dato.

3.1.1 Variazioni dell'architettura

Breve descrizione diconvAE stacked SAE DAE VAE

3.1.2 Applicazioni principali

La capacità degli Autoencoder di ricostruire l'input, quindi di mantenerne la struttura, privandolo di eventuale rumore o addirittura rimpiazzando il rumore con valori verosimili, si è rivelata utile in molti campi.

Quando il rumore incide notevolmente

In questo paper si usano gli AE per colorare immagini un bg

Anche in questo documento la rimozione del rumore sarà un tema centrale, l'applicazione nel nostro caso verrà analizzato nel dettaglio nei prossimi capitoli.

3.2 Convoluzioni e Convoluzioni Trasposte

3.3 Spazi Latenti

Capitolo 4

Valutazione Sperimentale

Capitolo 5

Risultati Ottenuti

Capitolo 6

Metodi Alternativi

Capitolo 7

Ulteriori Risultati

Capitolo 8

Conclusioni

Appendice A

Come si fanno le appendici

Le appendici si fanno con `\appendix` seguito da `\chapter{...}`

Appendice B

Esempi di Citazioni Bibliografiche

Pýrlå in [2] ha poi generalizzato i risultati di Bišker [1].

Il pacchetto `uniudtesi` carica automaticamente `hyperref`, che a sua volta rende “cliccabili” i riferimenti bibliografici nel documento elettronico.

Appendice C

Ambiente GNU/Linux (ad esempio Ubuntu)

Contributo di
Leonardo Taglialegne

Gli ambienti GNU/Linux contengono parecchi strumenti utili per la stesura di una tesi di laurea, in particolare segnaliamo:

- Kile
- KBibTeX

Il primo è un editor per il \LaTeX , che include una tabella dei simboli, la visualizzazione della struttura, evidenziazione del codice e simili comodità, e nelle ultime versioni fornisce una visualizzazione in anteprima dei risultati di compilazione.

Il secondo è uno strumento di ricerca, modifica ed inserimento di citazioni in formato BibTeX.

I pacchetti relativi (ed altri utili) si installano, su ambienti Debian e Ubuntu con: `sudo apt-get install kile kile-l10n kbibtex texlive-science texlive-math-extra texlive-lang-italian`

Bibliografia

- [1] J. Bišker, *On the elements of the empty set*. Mathematica Absurdica **132** (1999), 13–113.
- [2] U. Pърла, *Generalization of Bišker's theorem*. Paperopolis J. Math. **14** (2001), 125–132.