
Elenco delle figure

2.1	Struttura del Neurone	2
2.2	Architettura rete neurale convoluzionale	3
2.3	Esempio Database MNIST	6
2.4	Struttura Fashion-MNIST	7
2.5	Diagramma del processo di conversione utilizzato per generare il dataset Fashion-MNIST.	8
2.6	Struttura LIRA	9
2.7	Reti neurali a confronto	11
2.8	Test di accuratezza su F-MNIST	14
2.9	Layer architettura LeNet-5	14
2.10	Layer architettura AlexNet	15
2.11	Layer architettura Res-Net-56	15
2.12	Test di accuratezza su F-MNIST con architettura LeNet-5	16

Elenco delle tabelle

2.1 Sperimentazione del tasso di apprendimento	13
--	----

CAPITOLO 2

Background e Stato dell'arte

2.1 Background

2.1.1 Reti Neurali

Una **Rete Neurale** è un insieme di algoritmi che tentano di scoprire le relazioni sottostanti ai set di dati attraverso un processo che imita il funzionamento del cervello umano. Una rete neurale si riferisce a un sistema di neuroni.

Cos'è un neurone? Un neurone è una funzione matematica che riceve un input, elabora quell'informazione e restituisce un output.

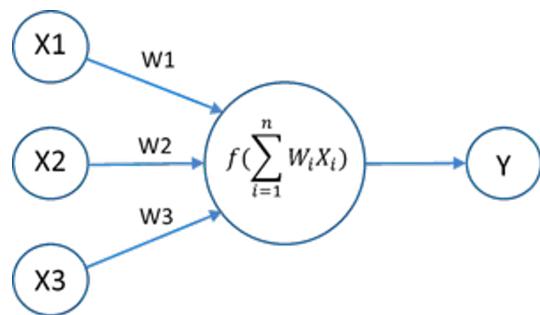


Figura 2.1: Struttura del Neurone

La (Figura 2.1) mostra un neurone con 3 input (X1,X2,X3), ad ogni input è associato un peso (w1,w2,w3), gli input con i relativi pesi vengono rangati nel neurone che poi restituirà l'output Y.

2.1.2 Convolutional Neural Network (CNN)

La rete neurale convoluzionale (CNN o ConvNet) è una classe di rete neurale artificiale (Versioni regolarizzate di percepitrone multistrato¹) che estraie proprietà dai dati di input e viene addestrata utilizzando un algoritmo di back-propagation² della rete neurale. Le CNN possono apprendere mappature complesse, ad alta dimensione e non lineari da un numero molto elevato di dati (immagini). [1]

Albawi et al. (2017) affermano che le reti neurali convoluzionali (CNN) sono una delle reti neurali profonde più popolari. [2]

Una CNN come si può vedere in (Figura 2.2) è composta da diversi livelli:

- **livello convoluzionale:** conserva le proprietà dell'immagine facendo scorrere una matrice più piccola (kernel o filtro) su di essa creando una mappa caratteristica;
- **livello di pooling:** viene utilizzato per mantenere gli aspetti più importanti, arrivando così ad un appiattimento della mappa delle caratteristiche;
- **livello completamente connesso:** connette ogni neurone di un livello ai neuroni dei livelli precedenti e successivi, ricavando così la matrice di input dal livello precedente, che viene levigata e inoltrata al livello di output in cui vengono effettuate le previsioni.

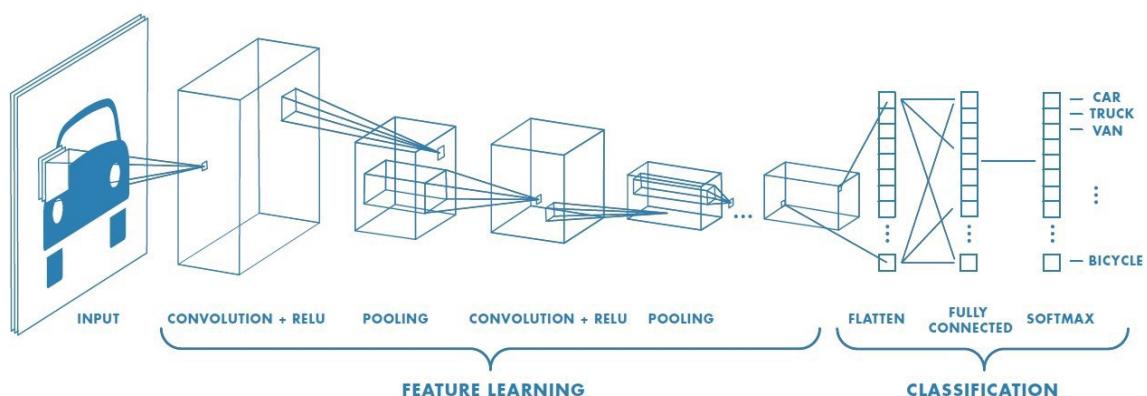


Figura 2.2: Architettura rete neurale convoluzionale

¹Modello di rete neurale artificiale che mappa un insieme di dati di input su un insieme appropriato di dati di output. Consiste di più livelli di nodi in un grafo diretto, dove ogni livello è completamente connesso al successivo

²Algoritmo che confronta il valore di uscita del sistema con il valore desiderato. Sulla base delle differenze (errori) così calcolate, l'algoritmo modifica i pesi sinaptici della rete neurale per far convergere gradualmente l'insieme dei valori di uscita al valore desiderato

Grazie a questa architettura, l'apprendimento richiede meno parametri e riduce la quantità di dati necessari per addestrare il modello. Le CNN si ispirano alle reti neurali a ritardo temporale, in cui i pesi sono condivisi nella dimensione temporale per ridurne il calcolo. Il successo dell'addestramento degli strati gerarchici fa sì che le CNN siano le prime architetture di deep learning di successo, come dimostrato da Liu et al. (2017).

Le CNN hanno mostrato prestazioni eccellenti nelle applicazioni di apprendimento automatico. Grazie ai loro vantaggi pratici hanno migliorato le prestazioni e la precisione nei sistemi in applicazioni come la **classificazione delle immagini**, la visione artificiale, l'elaborazione del linguaggio naturale (Albawi et al., 2017), la classificazione per età e il genere (Levi e Hassner, 2015), classificazione del testo (Lai et al, 2015), riconoscimento dell'espressione facciale (Mollahosseini et al, 2016), riconoscimento vocale (Abdel-Hamid et al, 2014) [2]

2.1.3 La classificazione delle immagini

Classificare le immagini significa assegnare etichette alle immagini date in input scegliendole da un insieme fisso di categorie. Tale classificazione ha diversi usi tra cui: progettazione di robot, identificazione di oggetti, auto a guida autonoma, elaborazione dei segnali stradali.

Le CNN vengono applicate a set di dati di grandi dimensioni per apprendere e riutilizzare le rappresentazioni delle immagini per la classificazione. Come mostrato da Wang e Xi (2015), le CNN offrono una migliore precisione nella classificazione delle immagini sul set di dati CIFAR-10³ rispetto ad altre reti neurali. [2]

Le CNN possono apprendere mappature complesse, ad alta dimensione e non lineari, da un numero molto elevato di dati (immagini). Inoltre, le CNN forniscono un'eccellente valutazione media dell'immagine. I principali vantaggi sono l'estrazione di caratteristiche salienti che non vengono mai modificate e la loro invarianza a: spostamenti, e distorsioni dei dati di input (immagini). [1] [2]

2.1.4 Classificazione delle immagini nell'ambito della moda

Uno dei problemi più difficili nella classificazione multiclasse è la classificazione della moda, in cui le etichette vengono assegnate alle immagini che caratterizzano i tipi di abbigliamento. La difficoltà di questo problema di classificazione multiclasse della moda è dovuta

³Dataset composto da 60000 immagini a colori 32x32 in 10 classi, con 6000 immagini per classe. Sono disponibili 50000 immagini di allenamento e 10000 di prova <https://www.cs.toronto.edu/~kriz/cifar.html>

alla ricchezza delle caratteristiche dell'abbigliamento e alla profondità della classificazione. Questa profondità complessa fa sì che etichette/classi diverse abbiano proprietà simili. [1]

I modelli di deep learning hanno fatto un enorme passo avanti nelle attività di classificazione delle immagini di visione artificiale. Nel deep learning, ci riferiamo spesso ad alcuni fattori nascosti come iperparametri, questo perché sono uno dei componenti più importanti.

Cosa sono gli iperparametri? Sono elementi di ottimizzazione che sono esterni a un modello, ma possono avere un grande impatto sul suo comportamento e le prestazioni del modello dipendono fortemente dalla scelta degli iperparametri corretti. [3]

2.1.5 MNIST e Fashion MNIST

Data la sua grande efficienza la rete neurale convoluzionale è stata applicata al problema della classificazione delle immagini. Per testare le prestazioni della CNN sono stati utilizzati i dataset MNIST e Fashion-MNIST. [2]

MNIST: il dataset MNIST, composto da 10 classi di cifre scritte a mano, è stato introdotto per la prima volta da LeCun et al. [1998] nel 1998. [4]

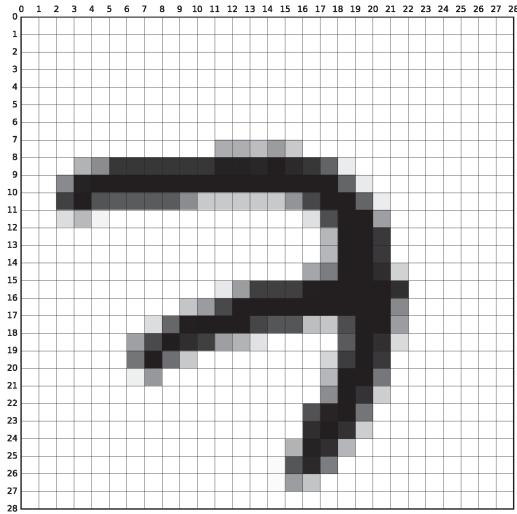
Il database MNIST contiene un totale di 70.000 istanze, di cui 60.000 per l'addestramento e il resto per il test. Questo database è composto da due diverse fonti: Special Database 1 del NIST⁴ [5] e Special Database 3 del NIST⁵. [6] I set di addestramento e test sono stati scelti in modo che lo stesso autore non fosse coinvolto in entrambi i set. Il set di formazione include esempi di oltre 250 autori. [7]

Le immagini originali sono state sottoposte a pre-elaborazione. Questa procedura ha comportato innanzitutto la normalizzazione delle immagini per farle rientrare in un riquadro di 20×20 pixel, preservando il rapporto d'aspetto. Poi è stato applicato un filtro anti-aliasing e le immagini in bianco e nero sono state trasformate in scala di grigi. Infine, è stato introdotto un padding vuoto per inserire le immagini in un riquadro più grande di 28×28 pixel, in modo che il centro di massa della cifra corrispondesse al suo centro. [7]

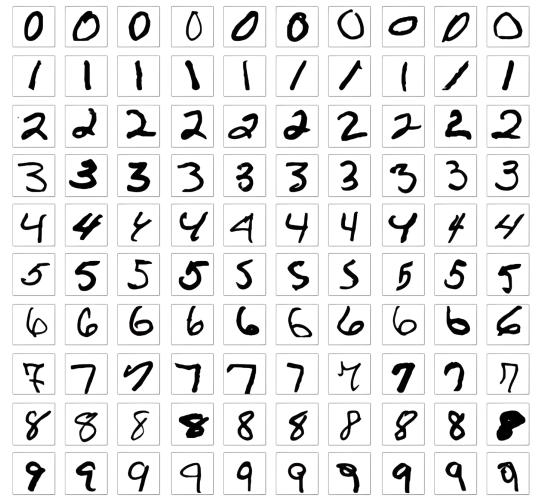
⁴Contiene 2.100 immagini a pagina intera di campioni di scrittura a mano stampati da 2.100 diversi studenti geograficamente distribuiti negli Stati Uniti

⁵Contiene i dati del campione, ovvero le informazioni raccolte dalle domande poste a un campione di tutte le persone e unità abitative, recuperato del Census Bureau

Un esempio di campione corrispondente alla cifra "7" è riportato nella (Figura 2.3a). La (Figura 2.3b) mostra un insieme più ampio di istanze appartenenti al set di addestramento.



(a) MNIST campione appartenente alla cifra 7



(b) 100 campioni dal set di addestramento MNIST

Figura 2.3: Esempio Database MNIST

A quel tempo, la rapida ascesa della tecnologia di deep learning e il suo potere erano imprevedibili. Nonostante il fatto che il deep learning possa fare molto oggi, il semplice set di dati MNIST è diventato il banco di prova più utilizzato per il deep learning, superando CIFAR-10 [Krizhevsky e Hinton, 2009] e ImageNet [Deng et al., 2009] attraverso le tendenze di Google Trends. [4] Infatti secondo Xiao, Rasul [4] e Baldominos [7], il motivo per cui MNIST è così popolare è dovuto alle sue dimensioni che permettono ai ricercatori di deep learning di verificare e prototipare rapidamente i loro algoritmi. A ciò si aggiunge il fatto che tutte le librerie di apprendimento automatico (ad esempio scikit-learn per la realizzazione quantistica del classificatore SFA [8]) e i framework di deep learning (ad esempio Tensorflow [9] e Pytorch [10]) forniscono funzioni di aiuto ed esempi pratici che utilizzano MNIST in modo immediato.

Fashion MNIST: si basa sull'assortimento presente sul sito web di Zalando⁶. Ogni prodotto di moda su Zalando ha una serie di immagini scattate da fotografi professionisti, che mostrano diversi aspetti del prodotto, ad esempio il fronte e il retro, i dettagli, l'aspetto con la modella e con un vestito. L'immagine originale ha uno sfondo grigio chiaro (colore

⁶La più grande piattaforma di moda online d'Europa <https://www.zalando.com>

esadecimale: #fdfdfdf) e viene salvata in formato JPEG 762x1000. L'immagine originale viene ricampionata con diverse risoluzioni, ad esempio grande, media, piccola, miniatura e minuscola. [4]

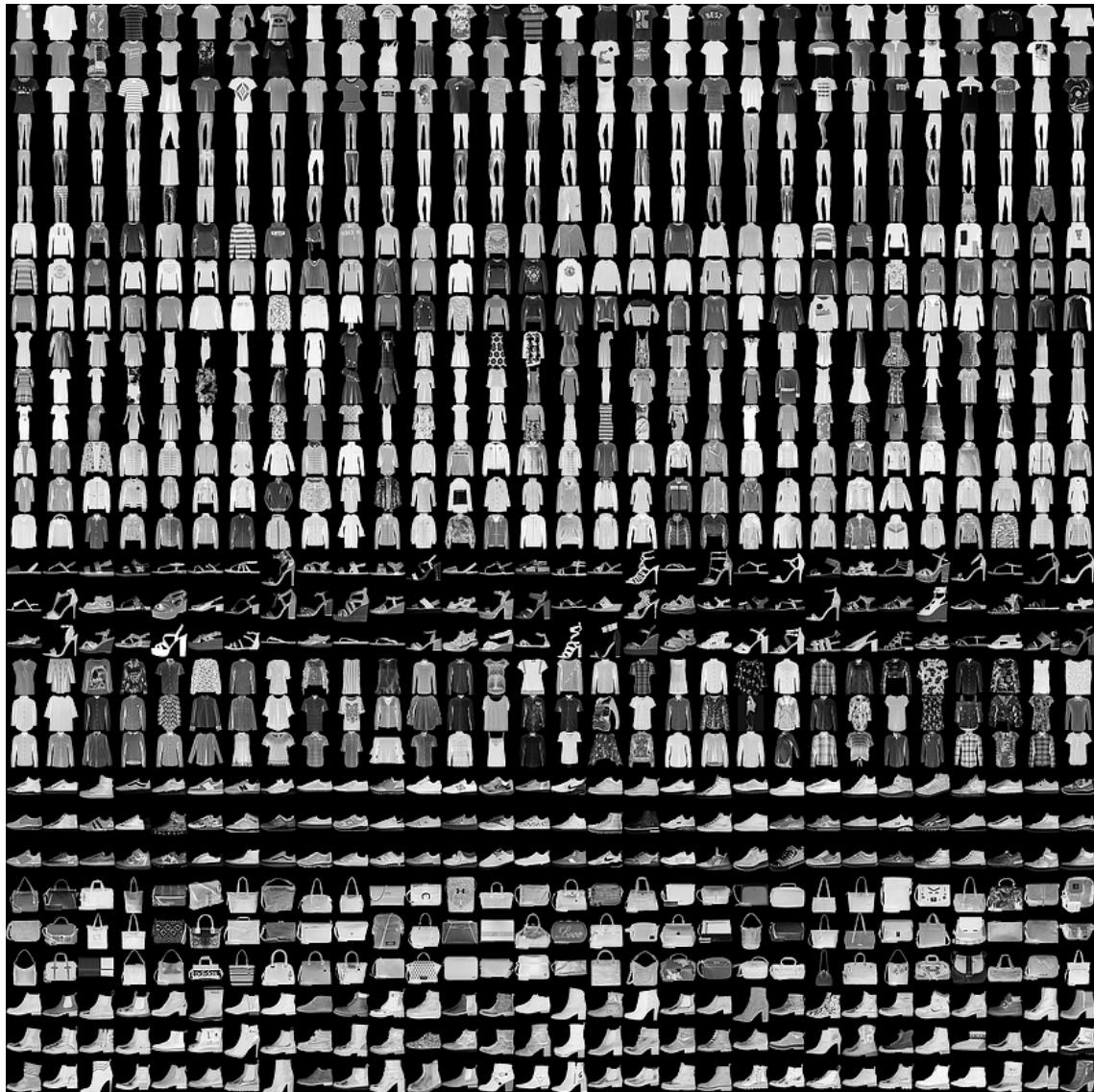


Figura 2.4: Struttura Fashion-MNIST

Fashion-MNIST è stato creato con 70.000 miniature di prodotti unici. Come mostrato nella (Figura 2.4) questi prodotti appartengono a diversi gruppi di genere come uomini, donne, bambini e persone neutre, inoltre ogni classe associata ad un capo di abbigliamento occupa tre righe. I prodotti bianchi in particolare non sono inclusi nel set di dati perché hanno un basso contrasto con lo sfondo.

La (Figura 2.5) mostra come le miniature (51x73) vengono inviate alla pipeline di conversione seguente:

1. Conversione dell'input in un'immagine PNG;
2. Ritagliare i bordi che sono vicini al colore dei pixel d'angolo. La "vicinanza" è definita dalla distanza entro il 5% dell'intensità massima possibile nello spazio RGB;
3. Ridimensionamento del bordo più lungo dell'immagine a 28 mediante sottocampionamento dei pixel, ossia saltando alcune righe e colonne;
4. Nitidezza dei pixel utilizzando un operatore gaussiano di raggio e deviazione standard pari a 1,0, con effetto crescente in prossimità dei contorni;
5. Estensione del bordo più corto a 28 e posizionamento dell'immagine al centro della tela;
6. Negare le intensità dell'immagine;
7. Conversione dell'immagine in pixel in scala di grigi a 8 bit;

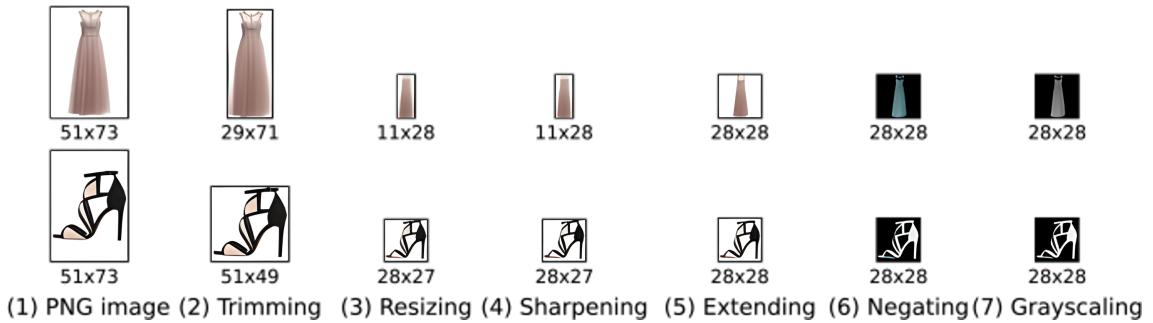


Figura 2.5: Diagramma del processo di conversione utilizzato per generare il dataset Fashion-MNIST.

2.2 Stato dell’arte

Questo capitolo illustra lo stato dell’arte e i lavori presentati in letteratura relativi alla classificazione di immagini di MNIST e Fashion-MNIST tramite CNN.

Architetture CNN per la classificazione di immagini

Molte architetture CNN sono state utilizzate per la classificazione delle immagini: LeNet, Alex Net, Google Net, VGGNet e ResNet. [1] Tutte queste architetture possono classificare e riconoscere correttamente le immagini. Le reti neurali sono state applicate anche all’apprendimento metrico con applicazioni nella stima della somiglianza delle immagini e nella ricerca visiva. Di recente sono stati pubblicati due set di dati. MNIST e Fashion-MNIST per la classificazione delle immagini utilizzando 70.000 immagini reali annotate. [1] Esamineremo brevemente i lavori svolti sul dataset Fashion-MNIST e MNIST.

Limited Receptive Area

Kussul e Baidyk (2004) [2] hanno proposto un classificatore neurale Limited Receptive Area (LIRA) per il riconoscimento delle immagini.

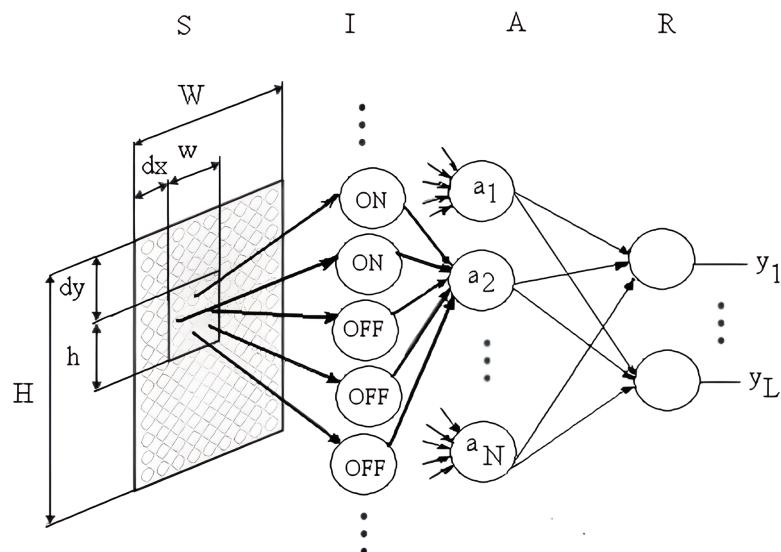


Figura 2.6: Struttura LIRA

Nella (Figura 2.6) viene mostrato come tale classificatore è costituito da 3 strati di neuroni:

- **Strato S:** neuroni sensore ;
- **Strato A:** neuroni associativi;

- **Strato R:** neuroni di risposta;

Lo strato S corrisponde all’immagine in input (per esempio l’immagine piatta di una cifra scritta a mano se si vuole considerare il set di dati MNIST), il secondo strato A corrisponde al sottosistema di estrazione delle caratteristiche. Il terzo strato R rappresenta l’uscita del sistema. Ogni neurone di questo strato corrisponde a una delle classi di uscita. Lo strato associativo A è collegato allo strato sensore S con connessioni scelte a caso e non addestrabili (sottostrato I). I pesi di queste connessioni possono essere pari a 1 (connessione positiva ON) o a -1 (connessione negativa OFF). L’insieme di queste connessioni può essere considerato come un estrattore di caratteristiche. Ciascun neurone dello strato A è collegato a tutti i neuroni dello strato R.

Kussal e Baidyk principalmente hanno applicato il classificatore LIRA per il riconoscimento della texture delle superfici metalliche [11]. La ragione per scegliere tale sistema basato sull’architettura delle reti neurali per questo compito è che tale sistema ha già dimostrato la sua efficacia nel riconoscimento delle immagini piatte grazie alle sue significative proprietà di adattabilità e robustezza alla varietà delle immagini. Infatti prima di procedere Kussal e Baidyk hanno testato il classificatore neurale LIRA nel compito di riconoscimento di cifre scritte a mano e il suo tasso di riconoscimento sul database MNIST è stato dello 0,55% [11], mostrando un’accuratezza del 99,41%, uno dei migliori risultati ottenuti su questo database.

ConvNet

Uno dei primi studi ha esaminato la capacità delle reti neurali profonde di ottenere risultati da record su set di dati supervisionati estremamente difficili (Krizhevsky et al., (2012) [3]). La rete contiene 5 livelli CNN e 3 livelli completamente connessi.

La (Figura 2.7a) mostra una normale rete neurale a 3 strati, mentre la (Figura 2.7b) mostra un ConvNet che dispone i suoi neuroni in tre dimensioni (larghezza, altezza, profondità), come visualizzato in uno dei livelli. Ogni livello di un ConvNet trasforma il volume di input 3D in un volume di output 3D di attivazioni neuronali. In questo esempio, il livello di input rosso contiene l’immagine, quindi la sua larghezza e altezza sarebbero le dimensioni dell’immagine e la profondità sarebbe 3 (canali: rosso, verde e blu).

Krizhevsky et al hanno ottenuto i migliori risultati mai registrati utilizzando uno dei più grandi ConvNet su un sottoinsieme del set di dati ImageNet [2]. Questa rete neurale include molte caratteristiche nuove e insolite come la non linearità di relu, le sovrapposizioni

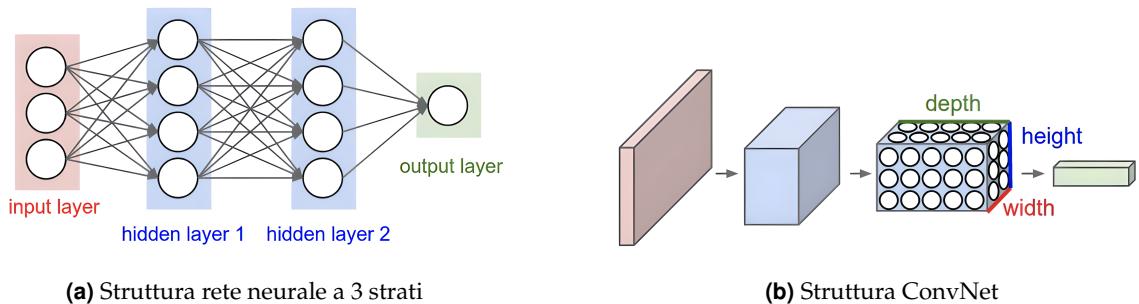


Figura 2.7: Reti neurali a confronto

di pooling, ecc. per migliorare le prestazioni e ridurre i tempi di allenamento. Sono stati utilizzati diversi metodi efficaci per ridurre l'over-fitting⁷ quali:

- **Early-stopping:** strategia per evitare il fenomeno del "rallentamento della velocità di apprendimento". Ciò significa che, oltre un certo punto, la precisione dell'algoritmo smette di migliorare o addirittura diminuisce a causa del rumore di allenamento. Dagli anni '90, è stato ampiamente utilizzato negli algoritmi iterativi, in particolare nelle reti neurali [12]. Se si interrompe l'apprendimento prima del punto si ha un under-fitting⁸, oltre si verifica un over-fitting. Per determinare esattamente dove interrompere l'apprendimento bisogna tener traccia dell'accuratezza sui dati di test mentre la rete viene addestrata interrompendo l'allenamento quando i dati del test non migliorano più in termini di precisione. Steve Lawrence e Lee Giles [13] hanno combinato l'early-stopping con l'algoritmo di back-propagation;
 - **Espansione dei dati di addestramento:** strategia ampiamente utilizzata e si è dimostrata efficace per migliorare le prestazioni di generalizzazione dei modelli in molte aree applicative, come il riconoscimento dei modelli e l'elaborazione delle immagini [12]. In effetti, nell'apprendimento automatico, l'algoritmo non è l'unica cosa che influisce sull'accuratezza della classificazione finale. In molti casi le prestazioni del classificatore possono essere influenzate in modo significativo dalla quantità e dalla qualità dei dati di addestramento, soprattutto nell'ambito dell'apprendimento supervisionato [12]. Parametri ben tarati consentono di raggiungere un buon equilibrio tra accuratezza e regolarità dell'addestramento, inibendo così l'effetto di over-fitting e di under-fitting. Per risolvere questo problema, si possono manipolare i dati esistenti per generarne di nuovi ampliando il set di allenamento (Ad esempio acquisire più dati di addestramento

⁷Rischio di sovrardattamento durante il processo di apprendimento induttivo

⁸Problema di apprendimento che si verifica quando la classificazione si basa su pochi parametri

[14] oppure produrne di nuovi sulla base della distribuzione del set di dati esistente [15]);

Hu et al. (2015) hanno proposto un’architettura CNN a 5 strati che è stata testata su vari set di dati di immagini iperspettrali per classificare direttamente le immagini nel dominio spettrale per migliorare le prestazioni. Il framework CNN, come Caffe, è stato utilizzato per ridurre i tempi di addestramento e di test e ha ottenuto un’accuratezza del 90% sul set di dati MNIST. [2]

Algoritmo SVM

Tang, Y. (2013) [2] ha dimostrato che la sostituzione dello strato softmax con SVM è utile per i compiti di classificazione. Gli esperimenti sono stati eseguiti utilizzando i set di dati MNIST e CIFAR-10. L’utilizzo di SVM ha migliorato l’accuratezza della convalida incrociata per la fase di test al 68,9% rispetto al 67,6% di Softmax. L’architettura proposta da Tang (2013) è stata emulata da Agarap (2017) [2] combinando una rete neurale convoluzionale (CNN) e una SVM lineare per la classificazione delle immagini sul set di dati MNIST. I risultati mostrano che l’accuratezza del test dei modelli CNN-SVM e CNN-Softmax sul set di dati MNIST è di circa il 99,04% contro il 99,23%.

Il dataset Fashion-MNIST

Il set di dati Fashion MNIST è stato presentato da Zalando Research⁹ (Xiao et al., 2017 [3]). Fashion-MNIST si propone come un sostituto diretto del tradizionale set di dati MNIST di cifre scritte a mano, considerato un punto di riferimento per le tecniche di apprendimento automatico, poiché condivide la stessa struttura, formato dell’immagine e dimensioni di train e test set. Tuttavia Fashion-MNIST pone un compito di classificazione più impegnativo rispetto ai semplici dati di MNIST.

La (Tabella 2.1) che segue, realizzata da Shivam et Amol [2] mostra il tasso di apprendimento della funzione di attivazione sigmoide da loro proposta per il modello CNN. Infatti la precisione sul dataset MNIST è superiore al 99% per la maggior parte dei casi a differenza del dataset Fashion-MNIST sintomo di una maggiore facilità del dataset.

⁹Lanciato nel 2016 è il dipartimento duraturo ed esplorativo dell’azienda che riunisce esperti interni, scienziati e sviluppatori <https://github.com/orgs/zalandoresearch/repositories>

Learning rate	Set di dati F-MNIST		Set di dati MNIST	
	Trining accuracy	Testing accuracy	Trining accuracy	Testing accuracy
0.01	87.50%	85.43%	98.69%	97.85%
0.02	84.50%	84.01%	99.68%	98.18%
0.001	92.59%	88.67%	99.73%	98.19%
0.002	92.93%	89.21%	99.75%	98.18%

Tabella 2.1: Sperimentazione del tasso di apprendimento

I membri delle comunità AI¹⁰, ML¹¹ e Data Science¹² adorano questo set di dati e lo usano come benchmark per convalidare i propri algoritmi. In effetti, MNIST è spesso il primo set di dati provato dai ricercatori affermando che se l’algoritmo non funziona su MNIST, non funzionerà affatto, tuttavia se funziona su MNIST, potrebbe ancora fallire su altri. [3]

Il team di sviluppo Zalando sta cercando di sostituire il set di dati MNIST originale con il proprio set di dati più complesso portandoci ad un nuovo scenario in cui testare un algoritmo di addestramento che abbia successo su F-MNIST significa realizzare un algoritmo che abbia successo su qualsiasi set di dati con margine di dubbio nullo. [3]

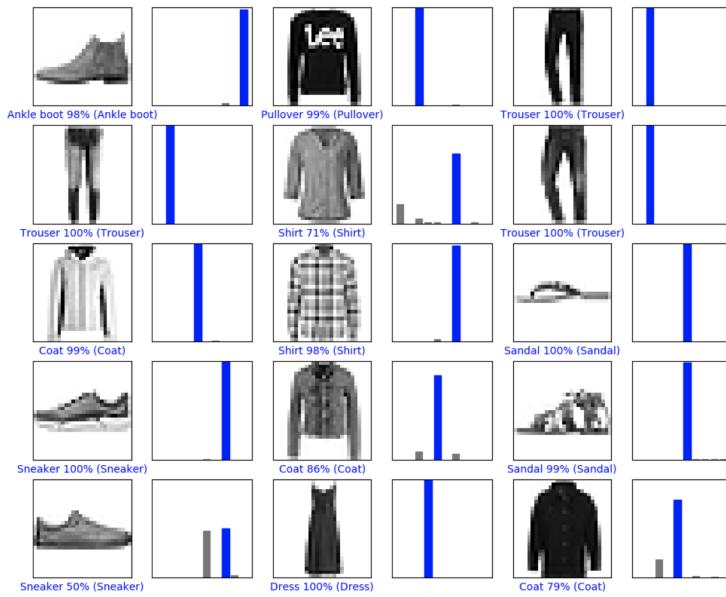
Nel lavoro di S. Bhatnagar, D. Ghosal e Kolekar M. H. (2017) [1], la categorizzazione Fashion-MNIST è stata condotta per classificare gruppi di immagini di articoli di moda. Sono stati dimostrati 3 diversi modelli ConvNet e sono state applicate le connessioni residue saltate e la normalizzazione in batch (BN) per facilitare e velocizzare il processo di apprendimento. Gli autori hanno ottenuto un’accuratezza del 92,54%. [2] [1]

Nella (Figura 2.8) che segue, viene mostrato nello specifico l’accuratezza (in termini di percentuale) nel riconoscimento di diversi capi di abbigliamento appartenenti a categorie differenti. L’accuratenza è massima per tutti i capi di abbigliamento fatta eccezione per alcune tipologie di scarpe e magliette.

¹⁰Intelligenza Artificiale: vasta area dell’informatica che associa l’intelligenza umana alle macchine

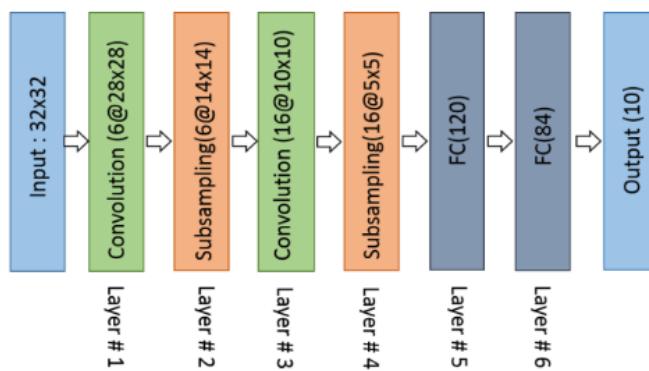
¹¹Machine Learning: area dell’informatica che si occupa dell’apprendimento automatico delle macchine

¹²Scienza multidisciplinare incentrata sulla scoperta di informazioni fruibili da grandi insiemi di dati grezzi (non strutturati) e strutturati

**Figura 2.8:** Test di accuratezza su F-MNIST

Manessi et Rozza (2018) [2] hanno introdotto due approcci per apprendere le diverse combinazioni di funzioni di attivazione di base, ovvero la funzione identità, ReLU e tanh. Gli approcci proposti sono stati confrontati con architetture ben note, ovvero:

- **LeNet-5:** rete neurale convoluzionale basata sulla struttura della corteccia sensoriale della corteccia visiva umana, simile alle macchine neurocognitive, e addestrata con un algoritmo di backpropagation. Si differenzia dalle reti neurali ordinarie per il dominio ricettivo locale e la condivisione dei pesi, che consentono a Lenet-5 di ridurre il numero di parametri nel processo di costruzione della rete e di accelerare il processo di apprendimento. Come mostrato in (Figura 2.9) la sua struttura prevede una serie di layer convoluzionali (Bloc verdi) alternati a livelli di pooling (Bloc arancioni). [16];

**Figura 2.9:** Layer architettura LeNet-5

- **AlexNet:** rete neurale convoluzionale simile a LeNet-5 tuttavia più larga e profonda e prevede una successione di diversi layer convoluzionali anziché alternarli a livelli pooling come mostrato in (Figura 2.10) [17];

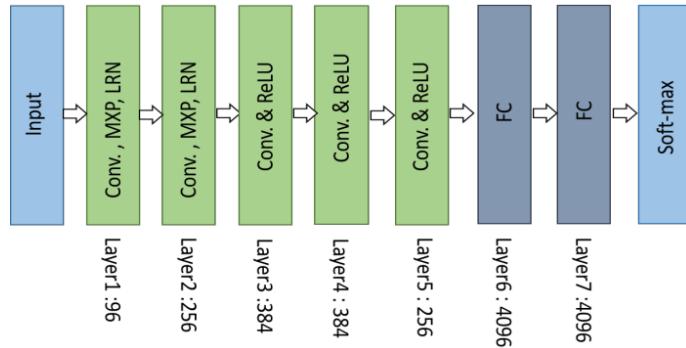


Figura 2.10: Layer architettura AlexNet

- **Res-Net-56:** rete neurale convoluzionale a 56 livelli utilizzata principalmente per trovare una soluzione a problemi complessi. L’intuizione alla base è l’aggiunta di più livelli, il cui numero varia a seconda della complessità del problema, che apprendono progressivamente funzionalità più complesse. Ad esempio, in caso di riconoscimento delle immagini, il primo strato può imparare a rilevare i bordi, il secondo strato può imparare a identificare le trame e allo stesso modo il terzo strato può imparare a rilevare oggetti e così via. Ma è stato scoperto che esiste una soglia massima per la profondità con il tradizionale modello di rete neurale convoluzionale [18]. La (Figura 2.11) mostra la sua struttura ridotta, per ottenerla nella sua completezza basta moltiplicare i layer per la quantità su di essi indicata;

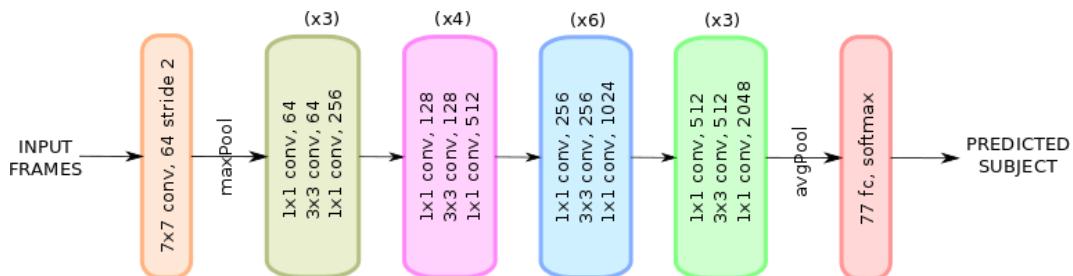


Figura 2.11: Layer architettura Res-Net-56

Per testare queste architetture sono stati utilizzati tre dataset standard (Fashion-MNIST, CIFAR-10 e ILSVRC-2012¹³). I risultati mostrano miglioramenti sostanziali nelle prestazioni complessive, principalmente l’aumento dell’accuratezza per AlexNet su ILSVRC-2012 di 3,01 punti percentuali rispetto ai test fatti fino a quel momento. Ottimi anche i risultati ottenuti dall’architettura LeNet-5 sul dataset Fashion-MNIST che ha mostrato un’accuratezza dell’88,9% (come si può vedere in Figura 2.12) considerando l’estrema complessità del dataset.

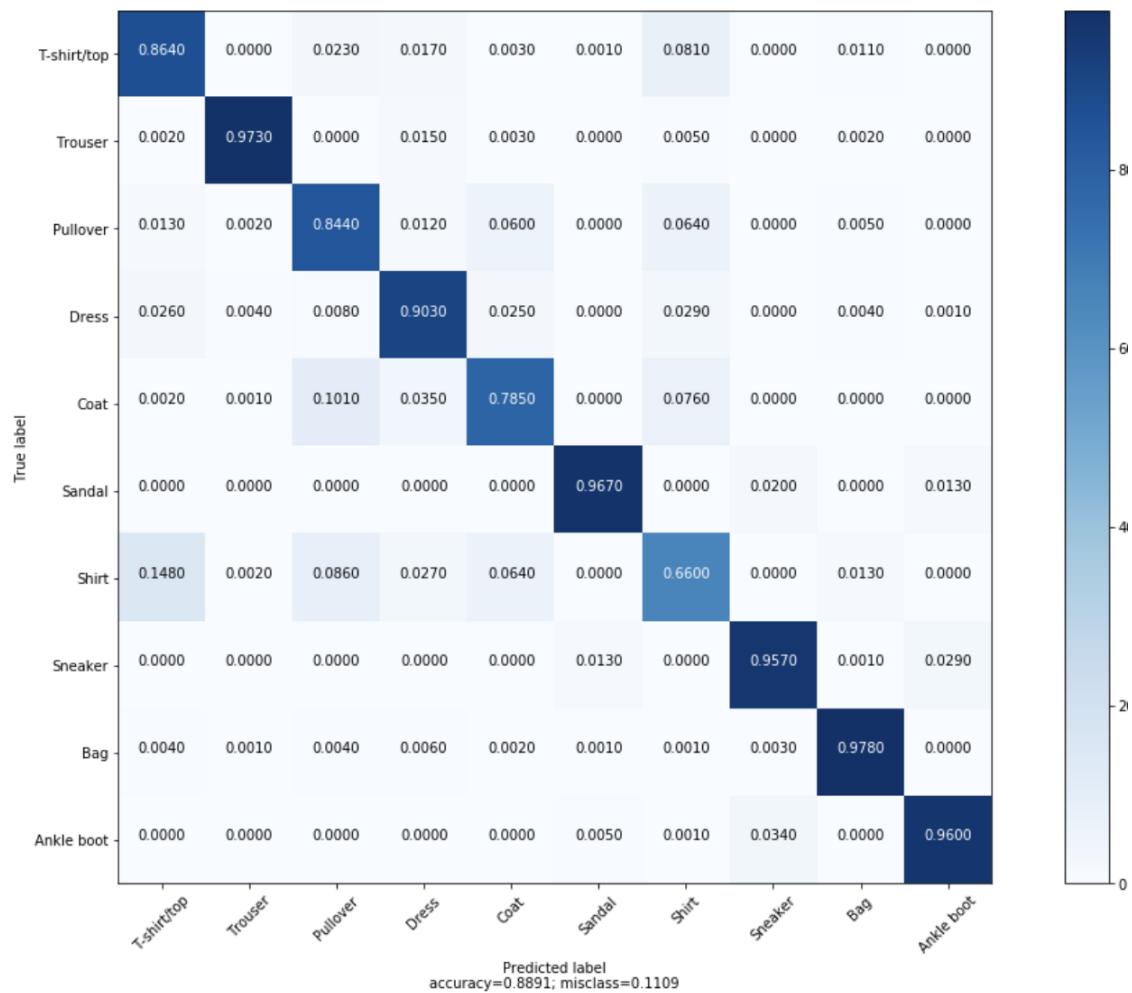


Figura 2.12: Test di accuratezza su F-MNIST con architettura LeNet-5

¹³ImageNet Large Scale Visual Recognition Challenge valuta gli algoritmi per il rilevamento degli oggetti e la classificazione delle immagini su larga scala <https://www.image-net.org/challenges/LSVRC/>

Michael McKenna ha proposto un modello che aggiunge e confronta le caratteristiche Sigmoid, ELU e ReLU di benchmark mancanti nel set di dati Fashion-MNIST. In primo luogo, le reti neurali feedforward multistrato non innovative, che mancavano a Fashion-MNIST, sono state considerate un benchmark. Successivamente, si è controllato l’efficacia della funzione di attivazione simultanee (rispetto a ELU, ReLU e Sigmoid). Fashion-MNIST ha molti benchmark, ma nessun benchmark di architettura superficiale, ed ELU è usato raramente rispetto alle reti convoluzionali, quindi l’obiettivo è nuovo. I risultati hanno mostrato che l’output era significativamente peggiore del benchmark convoluzionale e alcuni dei vantaggi di ELU e ReLU erano evidenti quando le reti venivano addestrate leggermente. [1]

Infine, Shuning Shen [1] ha utilizzato reti di memoria a breve termine per costruire un modello che utilizza il set di dati fashion MNIST per la classificazione delle immagini, riducendo il consumo di tempo e migliorando l’accuratezza predittiva del modello. I risultati hanno mostrato che il modello LSTM potrebbe adattarsi al set di dati con la massima precisione (88,26%).

Bibliografia

- [1] M. Kayed, A. Anter, and H. Mohamed, "Classification of garments from fashion mnist dataset using cnn lenet-5 architecture," in *2020 international conference on innovative trends in communication and computer engineering (ITCE)*, pp. 238–243, IEEE, 2020. (Citato alle pagine 3, 4, 5, 9, 13 e 17)
- [2] S. S. Kadam, A. C. Adamuthe, and A. B. Patil, "Cnn model for image classification on mnist and fashion-mnist dataset," *Journal of scientific research*, vol. 64, no. 2, pp. 374–384, 2020. (Citato alle pagine 3, 4, 5, 9, 10, 12, 13 e 14)
- [3] K. Greeshma and K. Sreekumar, "Hyperparameter optimization and regularization on fashion-mnist classification," *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 2, pp. 3713–3719, 2019. (Citato alle pagine 5, 10, 12 e 13)
- [4] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *arXiv preprint arXiv:1708.07747*, 2017. (Citato alle pagine 5, 6 e 7)
- [5] M. Garris, J. Blue, G. C, D. Dimmick, J. Geist, P. Grother, S. Janet, and C. Wilson, "Public domain optical character recognition," 07 1996. (Citato a pagina 5)
- [6] P. J. Grother, "Nist special database 19," *Handprinted forms and characters database, National Institute of Standards and Technology*, vol. 10, 1995. (Citato a pagina 5)
- [7] A. Baldominos, Y. Saez, and P. Isasi, "A survey of handwritten character recognition with mnist and emnist," *Applied Sciences*, vol. 9, no. 15, p. 3169, 2019. (Citato alle pagine 5 e 6)

- [8] I. Kerenidis and A. Luongo, "Quantum classification of the mnist dataset via slow feature analysis," *arXiv preprint arXiv:1805.08837*, 2018. (Citato a pagina 6)
- [9] F. Ertam and G. Aydin, "Data classification with deep learning using tensorflow," in *2017 international conference on computer science and engineering (UBMK)*, pp. 755–758, IEEE, 2017. (Citato a pagina 6)
- [10] D. Guidotti, F. Leofante, L. Pulina, and A. Tacchella, "Verification of neural networks: enhancing scalability through pruning," *arXiv preprint arXiv:2003.07636*, 2020. (Citato a pagina 6)
- [11] O. Makeyev, T. Baidyk, and A. Martín, "Limited receptive area neural classifier for texture recognition of metal surfaces," in *IFIP International Conference on Artificial Intelligence in Theory and Practice*, pp. 375–384, Springer, 2006. (Citato a pagina 10)
- [12] G. Raskutti, M. J. Wainwright, and B. Yu, "Early stopping and non-parametric regression: An optimal data-dependent stopping rule," *arXiv preprint arXiv:1306.3574*, 2013. (Citato a pagina 11)
- [13] R. Caruana, S. Lawrence, and C. Giles, "Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping," *Advances in neural information processing systems*, vol. 13, 2000. (Citato a pagina 11)
- [14] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1891–1898, 2014. (Citato a pagina 12)
- [15] K. Y. Yip and M. Gerstein, "Training set expansion: an approach to improving the reconstruction of biological networks from limited and uneven reliable interactions," *Bioinformatics*, vol. 25, no. 2, pp. 243–250, 2009. (Citato a pagina 12)
- [16] Y. Wang, F. Li, H. Sun, W. Li, C. Zhong, X. Wu, H. Wang, and P. Wang, "Improvement of mnist image recognition based on cnn," in *IOP Conference Series: Earth and Environmental Science*, vol. 428, p. 012097, IOP Publishing, 2020. (Citato a pagina 14)
- [17] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esen, A. A. S. Awwal, and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164*, 2018. (Citato a pagina 15)

- [18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016. (Citato a pagina 15)