Deep Learning per il Riconoscimento di Prodotti in Ambito *Retail*

Rocco Pietrini[1,3], Marina Paolanti[1,2], Valerio Placidi[1,3], Marco Contigiani[1,3], Luigi Di Bello[1,3], Primo Zingaretti[1], Adriano Mancini[1], Emanuele Frontoni[1,2]

[1] VRAI Vision Robotics and Artificial Intelligence Lab, Università Politecnica delle Marche, Department of Information Engineering (DII), Ancona, Italy,

[2] University of Macerata, Department of Political Sciences, Communication and International Relations, Macerata, Italy,

[3] Grottini Lab, Via Santa Maria in Potenza, 62017 Porto Recanati, Italy

{marina.paolanti,emanuele.frontoni}@unimc.it, {p.zingaretti, a.mancini}@staff.univpm.it, {rocco.pietrini,valerio.placidi,marco.contigiani,luigi.dibello}@grottinilab.com

Abstract

L'obiettivo del progetto è l'immplementazione di una *pipeline* per il riconoscimento di prodotti presenti su uno scaffale, a livello di singolo codice EAN a partire da una foto dello scaffale stesso. La pipeline si compone di una prima rete neurale che si occupa di effettuare la *detection* dei singoli prodotti presenti nello scaffale (la rete è stata realizzata da Goldman et al.) e una seconda rete, progettata e realizzata in collaborazione con Grottini Lab Srl che ha il compito di associare alla singola immagine realizzata dalla prima rete, un vettore di *embedding*, che ne descrive le *feature* distintive. Per svolgere questo compito attualmente non esistono dataset in grado di soddisfare i nostri requisiti poiché non presentano un livello di dettaglio così granulare (EAN). Per questo motivo, è stato realizzato interamente un nuovo dataset in collaborazione con l'azienda Grottini Lab e con il VRAI - Vision, Robotics and Artificial Intelligence - del dipartimento di Ingegneria dell'Informazione dell'Università Politecnica delle Marche.

1 Introduzione

In ambito *retail*, il riconoscimento automatico di prodotti riveste un ruolo fondamentale per migliorare la loro gestione da parte di venditori e produttori, con conseguente miglioramento e soddisfacimento dell'esperienza di acquisto dei clienti. In questo ambito, il nostro progetto ha come scopo il riconoscimento dei prodotti per ottenere una migliora gestione del planogramma (o la sua rilevazione automatica), che rappresenta la disposizione dei prodotti in uno scaffale, posizionati all'interno di un punto vendita, al fine di aumentarne le prestazioni e, conseguentemente, gli acquisti da parte dei clienti.

Solitamente, i planogrammi sono progettati direttamente dalla direzione centrale del *retailer* e vengono inviati ai singoli punti vendita. Essi sono realizzati tenendo conto di diversi fattori come lo spazio disponibile nello scaffale e i requisiti di rifornimento. Infatti, un planogramma può essere progettato, ad esempio, affinché un certo prodotto sia il meno possibile *out-of-stock* al fine di ridurre al minimo i mancati incassi. Non sempre i planogrammi sono rispettati dai singoli punti vendita, quindi il monitoraggio della conformità dei planogrammi rappresenta una sfida per i negozianti: che devono garantire la conformità dei planogrammi con quelli realizzati dalla sede centrale. A volte, non vengono rispettati poiché, ad esempio, i consumatori locali hanno delle esigenze diverse oppure semplicemente sono presenti errori (nell'implementazione o causati da errati riposizionamenti da parte dei clienti). Disattendere le direttive porta ad una bassa conformità del planogramma e alla conseguente incongruenza dei dati eventualmente raccolti in uno scaffale tramite altri strumenti, poiché non si ha più conoscenza della corretta posizione dei prodotti. Infatti, se alcuni dati sono indissolubilmente legati al planogramma (si pensi ad esempio all' analisi sul tempo passato dai clienti davanti un certo scaffale) essi non saranno più utilizzabili e, per questo motivo, avere scaffale non conformi al planogrammi non è accettabile per un punto vendita.

L'azienda Grottini Lab Srl[1], promotrice di questo progetto, si è posta l'obiettivo di semplificare notevolmente la gestione dei planogrammi nei punti vendita. L'idea è di identificare dalla foto dello scaffale cosa è presente e in che posizione esatta; se si ha un planogramma *master* (che può essere un'immagine oppure un planogramma testuale fornito dal retailer) viene realizzato il confronto per valutarne la conformità, altrimenti si usa tale immagine solamente per rilevare i prodotti, ignorando la verifica della conformità con il planogramma. Attualmente, questa attività viene fatta quasi del tutto in maniera manuale, inviando periodicamente il personale preposto al controllo nei punti vendita, quindi è ovvio che l'automatizzazione di tutto il processo consentirebbe una sua velocizzazione e la conseguente riduzione di costi ed errori umani. La verifica manuale di ogni singolo prodotto presente in uno scaffale è un'operazione estremamente *time consuming* poiché i codici a barre sono stampati in posizioni differenti (tra *packaging* di diversi prodotti) ed è quindi richiesto del tempo, non trascurabile per grandi numeri, per la loro ricerca e acquisizione. Alcune catene posizionano dei codici a barre nel cartellino dei prezzi ma essi possono riferirsi a codici interni invece che all'EAN e alcune volte accade che i prodotti non sono posizionati correttamente e i cartellini dei

[1] https://www.grottinilab.com

prezzi potrebbero riferirsi a prodotti diversi, per tale ragione ogni singolo prodotto va preso dallo scaffale e ne va scansionato il codice a barre (EAN). Infine, alcuni retailer hanno adottato *tag RFID*[2] nelle etichette dei loro prodotti ma tale soluzione risulta svantaggiosa sia da un punto di vista economico che di accuratezza in quanto le onde radio possono essere bloccate da altri oggetti o interferire tra di loro [Santra e Mukherjee, 2019].

Il nostro lavoro ha l'obiettivo di proporre un approccio capace di riconoscere i prodotti presenti su uno scaffale, a livello di singolo EAN, a partire da una foto dello scaffale acquisita con dispositivi mobile (smartphone, tablet) o fissi, superando gli attuali limiti imposti dagli algoritmi di *Computer Vision* come SIFT e rendendosi indipendente dalla sensoristica. In particolare, si vuole descrivere una possibile *pipeline* composta dall'app Store Audit di Grottini Lab, la rete neurale di Goldman [Goldman *et al.*, 2019] (che effettua la *detection* di prodotti a partire dall'immagine di uno scaffale) e una nuova rete con il compito di assegnare un vettore di *embedding* (ad ogni immagine in *output* alla rete precedente). Quest'ultimo è usato per inferire il giusto EAN valutando la *cosine similarity* con tutti i vettori di *embedding* di un dataset adibito al confronto, che sarà l'intero dataset. Bisogna considerare, inoltre, che il problema presenta numerose difficoltà e, infatti, una prima problematica può essere l'eccessiva somiglianza di prodotti che hanno EAN diversi oppure problemi relativi all'illuminazione ambientale, al punto di vista di chi scatta le foto, alle occlusioni, alla qualità stessa delle fotografie. L'intera pipeline, sviluppata in ambiente cloud AWS è illustrata in figura 1.

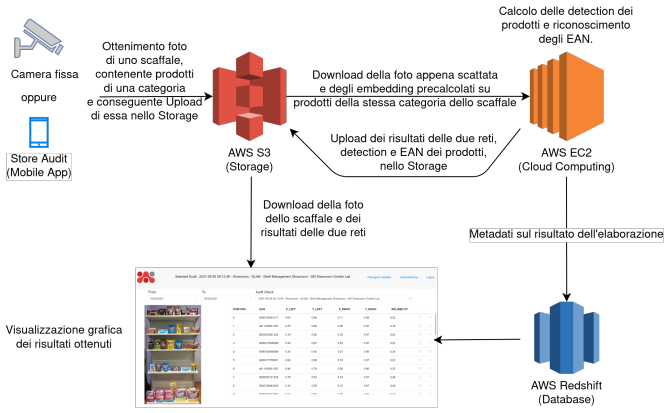

Figura 1: Flusso Logico

2 Materiali e metodi

Questo lavoro intende risolvere una problematica non particolarmente studiata nell'ambito del *Deep Learning*, quindi, non esistono *dataset* preesistenti e liberamente accessibili adatti allo scopo, contenenti cioè un numero elevato di immagini ottenute da scatti (in condizioni reali), realizzati con *smartphone*, di prodotti presenti negli scaffali di vari punti vendita ed etichettati con il giusto EAN.

[2]Radio Frequency IDentification

Si è proceduto quindi alla raccolta di un nuovo dataset di immagini di prodotti, etichettati con i corrispettivi EAN. Per questo scopo, si è proceduto a scattare le foto di tutti gli scaffali presenti in un punto vendita della catena Acqua & Sapone, di un punto vendita Oasi (Gruppo Gabrielli) e di un punto vendita Si con Te (CE.DI.MARCHE) oltre al reparto cioccolate di un Iper. Insieme a queste immagini sono stati raccolti gli EAN dei prodotti presenti nei corrispettivi scaffali tramite uno scanner di codici a barre. In particolare, sono state raccolte queste immagini: 176 al punto vendita Acqua&Sapone, 333 al punto vendita Oasi, 160 al punto vendita Si con Te e 4 al punto vendita Iper. Un esempio del dataset è riportato in figura 2.

(a) EAN: 8002910057701

(b) EAN: 7640110704721

(c) EAN: 8002190002644

(d) EAN: 4005900694843

Figura 2: Esempio del nostro dataset

Successivamente, è stata utilizzata la rete neurale [Goldman *et al.*, 2019] per effettuare la *detection* dei prodotti presenti nelle immagini, ottenendo numerose immagini di singoli prodotti contenuti nell'immagine di uno scaffale. Tale rete risolve il compito particolarmente complesso di rilevare, in maniera accurata, i singoli prodotti in scaffali densamente popolati e, allo stato dell'arte attuale, risulta essere una delle più efficienti ed efficaci. Inoltre, la rete è stata allenata con il dataset SKU110K, composto di *11,762* immagini di scaffali densamente popolati di prodotti di migliaia di punti vendita in tutto il mondo, dagli Stati Uniti all'Asia orientale, per un totale di $1.74 * 10^6$ *bounding box* [3].

In fase di creazione del dataset i *bounding box* realizzati da tale rete non sempre coincidono con le nostre necessità (individua alcuni falsi positivi come cartellini dei prezzi, oppure suddivide un unico prodotto in più prodotti) e, per questo motivo, è stato necessario creare un annotatore, realizzato in *Python* e dotato di interfaccia grafica realizzata con *tkinter*[4], che ci permettesse di creare manualmente dei *bounding box* personalizzati per poter realizzare il dataset con le migliori

[3]https://retailvisionworkshop.github.io/detection_challenge_2020/

[4]https://docs.python.org/library/tkinter.html

immagini possibili e che, infine, ci permettesse di associare, ad ogni immagine, il giusto EAN. Il risultato finale è stato l'ottenimento di un *dataset* composto di $35'802$ immagini di $14'426$ EAN distinti, divisi in 57 categorie. In media, di ogni EAN sono stati raccolti, quindi, circa $2,5$ immagini. Inoltre, ad ogni immagine, è stata associata una categoria derivante da un albero delle categorie messo a punto da GS1 nel 1999 come risultato del lavoro di un gruppo di aziende che hanno voluto rispondere all'esigenza di avere una classificazione merceologica comune per i prodotti[5]. Per i nostri scopi, l'annotazione si ferma al secondo livello di questo albero in quanto ritenuto un livello di granularità sufficiente. In fase di creazione del *dataset* e di etichettamento delle immagini, le categorie del secondo livello sono state convertite in numeri. La categorizzazione è nella forma *xxyy* dove *xx* rappresenta il primo livello dell'albero delle categorie e *yy* il secondo livello. Questa categorizzazione è utile in fase di inferenza in quanto è possibile effettuare i confronti solo su un sottoinsieme del *dataset*, aumentando quindi le *performance* temporali a causa di un minor numero di confronti da realizzare e riducendo gli errori.

2.1 Rete neurale

Poiché i prodotti nei vari punti vendita variano continuamente sia in numero che in tipologia, non si è potuto utilizzare una rete di classificazione con un numero fisso di classi altrimenti, per ogni nuovo articolo o variazione di *packaging* di uno preesistente, si sarebbe dovuto ri-addrestrare la rete di classificazione. La rete realizzata è quindi indipendente dal numero delle classi.

Il nostro approccio, quindi, vuole associare il giusto EAN all'immagine del prodotto tenendo conto della similarità tra le immagini, rendendosi indipendente dal numero delle classi e dalle modifiche ai *packaging*. Comunque, il riconoscimento di un nuovo prodotto richiede che all'interno del dataset usato per il confronto degli *embedding* sia presente la nuova immagine con il nuovo *packaging*, con il corrispettivo EAN. Infatti, la rete presentata genererà, per ogni immagine, un vettore di *embedding* che sarà poi confrontato, tramite calcolo della *cosine similarity*, con gli *embedding* delle immagini usate come dataset di confronto e dotate di *label* (l'EAN). L'approccio utilizzato per la risoluzione del nostro problema è fortemente ispirato al lavoro di [Schroff *et al.*, 2015], spostando il dominio applicativo, dal riconoscimento di volti all'individuazione degli EAN dei prodotti. In particolare, si è deciso di utilizzare la rete neurale MobileNetV2 [Sandler *et al.*, 2018] (senza strato finale di classificazione), seguita da uno strato con dimensionalità di 256. L'output della rete sarà, quindi, un *embedding* di dimensione 256 su cui è stato inoltre applicata la normalizzazione L2 come in *FaceNet*. Un esempio è riportato in figura 3.

Una volta allenata la rete, per calcolare l'immagine più simile a quella presa in *input* si calcola la *cosine similarity* tra l'*embedding* dell'immagine di *input* e l'*embedding* di ciascuna immagine utilizzata per la verifica della similarità. La *cosine similarity* tra due generici vettori numerici A e B è rappresentata dalla seguente formula ed è un numero compreso tra -1 e $+1$ dove valori prossimi a $+1$ indicano una più alta somiglianza (+1 corrisponde a vettori uguali):

$$cosine\ similarity = \frac{A \cdot B}{\|A\| \, \|B\|} \tag{1}$$

L'EAN dell'immagine con la più alta *cosine similarity* sarà (idealmente) l'EAN dell'immagine in input alla rete. L'allenamento della rete avviene attraverso delle triplette composte da un'immagine detta anchor, un'immagine positiva cioè rappresentante lo stesso EAN e un'immagine negativa (EAN differente). La rete cercherà di minimizzare la distanza tra gli embedding dell' anchor e della positiva e massimizzare la distanza tra l'anchor e la negativa. Una particolare scelta effettuata per la fase di training è stata quella di calcolare le triplette online e non offline e questo è derivato anche dalle conclusioni di [Hermans *et al.*, 2017]. Essi, infatti, mostrano come l'utilizzo di triplette calcolate online possa aumentare di molto l'accuratezza del modello e ridurre i tempi di allenamento.

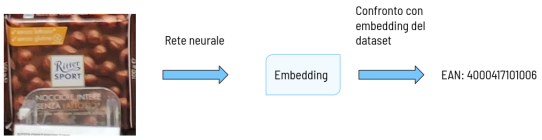

Figura 3: Gerazione del vettore di embedding

3 Risultati sperimentali

La rete neurale è stata implementata tramite il framework *Tensorflow* con il linguaggio di programmazione *Python*. Durante l'addestramento, sono stati utilizzati i seguenti parametri: la rete adibita all'estrazione di *feature* è la MobileNetV2 (decisa in fase sperimentale perché permette di ottenere migliori prestazioni), la dimensione dell'*embedding* è pari a 256, la dimensione di ogni immagine è 224x224 a 3 canali, la *loss* è la *Triplet Hard Loss* con margin di 1.0 (*soft margin*), l'ottimizzatore della *loss* è Adam [Kingma e Ba, 2017] con Learning Rate di 0.001 e l'allenamento è avvenuto in 80 epoche e, infine, la *batch size* è pari a 128. Sono stati, inoltre, caricati i pesi di *ImageNet* [Deng *et al.*, 2009], applicando la tecnica del *transfer learning* per velocizzare e migliorare l'apprendimento ed è stato anche utilizzato l'*early stopping* per evitare l'*overfitting* andando a valutare, per ogni epoca, l'andamento della *validation loss* e salvando i pesi che avevano la *validation loss* minore. Il dataset è stato diviso in *training set* e *validation set* con percentuali rispettivamente dell'80% e del 20%. Inoltre, prima di effettuare questa operazione di divisione, sono stati rimossi dal dataset totale 196 EAN (357 immagini di 17 categorie) presenti nel dataset del punto vendita Acqua&Sapone e presenti negli altri punti vendita: questi EAN isolati li abbiamo utilizzati come *test set* e sono, quindi, immagini che la rete neurale non ha mai visto, simulando una situazione reale. Infine, l'intero nuovo dataset è stato utilizzato per realizzare i confronti cioè, in fase di inferenza (con una immagine nuova), ogni immagine prodotta

[5]https://gs1it.org/migliorare-processi/relazione-industria-distribuzione-best-practice-ecr/albero-categorie-classificazione-condivisa-prodotti/

dalla rete di Goldman [Goldman *et al.*, 2019] è data in *input* alla nostra rete che restituisce in *output* un vettore di *embedding*. Per ognuno di questi vettori viene calcolata la *cosine similarity* tra tutti gli *embedding* del dataset adibito al confronto al fine di individuare il vettore (che è associato ad un EAN) con la più alta *cosine similarity* e, conseguentemente, l'immagine più simile. In questo modo si può ottenere l'EAN.

L'accuratezza è calcolata nel seguente modo:

$$Accuratezza(set) = \frac{\#immaginicorrettoEAN}{\#\ set} \quad (2)$$

3.1 Confronto tra *backbone*

Prima di decidere di utilizzare la rete MobileNetV2 come *backbone* (la rete che estrae le *feature*), abbiamo eseguito dei test di confronto valutando le prestazioni in termini di accuratezza. Le reti testate sono state la MobileNetV2, MobileNetV3 Large e Small [Howard *et al.*, 2019] e Xception [Chollet, 2017]. Nella Tabella 1 sono riportate le percentuali di accuratezza TOP1, TOP5 e TOP10 di ogni *backbone* testata. Per avere un confronto coerente, sono stati utilizzati gli stessi dati e parametri durante le diverse sessioni di allenamento. Il dataset di test contiene 357 immagini (196 EAN) di 17 categorie di un punto vendita Acqua&Sapone ed è stato usato tutto il dataset raccolto per il confronto (escluse, ovviamente, le 357 immagini).

Accuratezza	**TOP1** (%)	**TOP5** (%)	**TOP10** (%)
MobileNetV2	43.4	69.2	75.6
MobileNetV3Small	41.1	60.5	70.5
MobileNetV3Large	40.6	55.7	64.4
Xception	18.7	36.13	42.5

Tabella 1: Confronto tra le reti usate come *backbone*

3.2 Risultati ottenuti in un contesto reale

Facendo l'analisi solo su due categorie esemplificative (delle 17 categorie del dataset di test) usando la rete con MobileNetV2 come *backbone*, si ottengono i seguenti valori di accuratezza:

- "Cura persona/Igiene personale": **TOP1**: 50.0%, **TOP5**: 68.42%, **TOP10**: 76.31%
- "Cura persona/Igiene orale": **TOP1**: 35.41%, **TOP5**: 68.75%, **TOP10**: 79.16%

Il corretto EAN è, almeno due volte su tre, contenuto nella TOP5. Inoltre, la categoria "Cura persona/Igiene orale" presenta molti articoli appesi sui ganci e non appoggiati direttamente ai ripiani, complicando ulteriormente il riconoscimento a causa delle variazioni di angolatura del prodotto. Per migliorare le prestazioni della rete neurale proposta, si può pensare di utilizzare le informazioni della TOP5 o TOP10 per cercare di individuare il corretto prodotto quando essi risultano essere molto simili. Infatti, nella maggior parte dei casi il prodotto corretto risulta essere nella TOP10. Un ulteriore aumento del dataset raccolto, con conseguente ri-allenamento della rete e aumento degli *embedding* nel dataset di confronto, può portare ad ottenere risultati migliori così come l'utilizzo di diverse tecniche di Intelligenza Artificiale che saranno sviluppate in futuro dalla comunità scientifica. In figura 4 viene infine mostrata una dashboard dei risultati dell'analisi di uno scaffale.

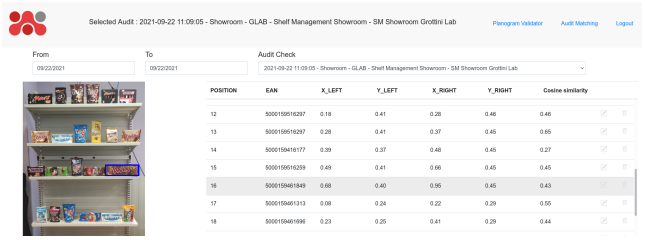

Figura 4: Dashboard di visualizzazione

Riferimenti bibliografici

[Chollet, 2017] François Chollet. Xception: Deep learning with depthwise separable convolutions, 2017.

[Deng *et al.*, 2009] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, e Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[Goldman *et al.*, 2019] Eran Goldman, Roei Herzig, Aviv Eisenschtat, Jacob Goldberger, e Tal Hassner. Precise detection in densely packed scenes. In *Proc. Conf. Comput. Vision Pattern Recognition (CVPR)*, 2019.

[Hermans *et al.*, 2017] Alexander Hermans, Lucas Beyer, e Bastian Leibe. In defense of the triplet loss for person re-identification, 2017.

[Howard *et al.*, 2019] Andrew Howard, Mark Sandler, Grace Chu, Liang-Chieh Chen, Bo Chen, Mingxing Tan, Weijun Wang, Yukun Zhu, Ruoming Pang, Vijay Vasudevan, Quoc V. Le, e Hartwig Adam. Searching for mobilenetv3, 2019.

[Kingma e Ba, 2017] Diederik P. Kingma e Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[Sandler *et al.*, 2018] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, e Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[Santra e Mukherjee, 2019] Bikash Santra e Dipti Prasad Mukherjee. A comprehensive survey on computer vision based approaches for automatic identification of products in retail store. *Image and Vision Computing*, 86:45–63, 2019.

[Schroff *et al.*, 2015] Florian Schroff, Dmitry Kalenichenko, e James Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, giu 2015.