Acquisizione di un dataset di immagini di frutta per confrontare le prestazioni di reti neurali convolutive profonde per la classificazione delle immagini

Vittorio Sala
iMAGE S s.p.a
vittorio.sala@imagesspa.it

Abstract

Negli ultimi anni la ricerca ha messo a disposizione degli utenti finali molte reti neurali convolutive per la classificazione di immagini. Il primo parametro da valutare nella scelta di una rete per una applicazione di machine vision industriale è sicuramente l'accuratezza che tuttavia non risulta prevedibile a priori in quanto combinazione dell'architettura della rete, della scelta dei parametri iniziali e del dataset. In questo lavoro si è creato un dataset di immagini di frutta con lo scopo di confrontare le prestazioni di diverse reti neurali convolutive profonde in termini di risorse computazionali richieste, tempi di training e di inferenza.

1 Introduzione

Lo scopo di questo lavoro è confrontare le performance di reti neurali convolutive allo stato dell'arte utilizzate per classificare immagini. Le reti scelte per il confronto sono quelle che hanno ottenuto i migliori risultati nella competizione ImageNet Large Scale Visual Recognition Challenge, e in particolare le architetture Resnet [He *et al*., 2016] e VGG [Simonyan e Zisserman, 2014]. Tali reti sono state confrontate con quelle incluse nella libreria di machine vision industriale Halcon [Richter e Streitferdt, 2018] sulla base degli intervalli di tempo richiesti nelle fasi di training, di inferenza e dell'utilizzo di risorse hardware. Allo scopo è stato creato un dataset di immagini di frutta [Muresan e Olten, 1985].

2 Preparazione del dataset

La scelta di acquisire un nuovo dataset di frutta è giustificata dal desiderio di avere a disposizione immagini con risoluzione maggiore di quelle disponibili in dataset acquisiti in precedenza [Muresan e Olten, 1985] allo scopo di migliorare l'accuratezza di classificazione. Si è quindi scelto di utilizzare una telecamera a colori IDS modello UI-3240CP-C-HQ con risoluzione pari a 1280x1024, equipaggiata con ottica per sensori da 1" con apertura 1:1.4 e lunghezza focale 12 mm. Il diaframma è stato regolato su F4, la distanza di lavoro utilizzata è pari a 50 cm e il campo inquadrato risulta pari a 210 mm x 170 mm. Il sistema ottico è stato installato coassialmente con un illuminatore a luce bianca diffusa quadrato di lato 300 mm installato alla medesima distanza di lavoro. Il piano di lavoro è stato coperto con carta comune bianca, utilizzata per il bilanciamento del bianco della telecamera. I campioni scelti sono simulacri di frutta. Sono state utilizzate 10 mele rosse, 10 mele rosa, e 8 pesche. La colorazione esterna dei frutti e la dimensione variano in modo importante all'interno della singola classe, pur mantenendo una trama comune. Per ogni campione sono state acquisite un centinaio di immagini in diverse posizioni e orientazioni ottenute riposizionando il campione manualmente ad ogni acquisizione. Sono state acquisite complessivamente 800 immagini per classe. Le immagini sono poi state elaborate con la libreria di machine vision Halcon. In particolare, i singoli frutti sono stati segmentati con tecniche di blob detection impostando una soglia sul canale blu, inscritti in un quadrato e ritagliati. I ritagli sono stati poi riscalati a 224 x 224 pixel, valore standard per le reti con architettura Resnet [He *et al*., 2016] e VGG [Simonyan e Zisserman, 2014]. Sono state poi estratte casualmente 400 immagini per classe per il training delle reti, 200 per la validazione e 200 per il test. Una immagine di esempio per classe è riportata in figura 1.

Figura 1: Da sinistra a destra: mela rossa, mela rosa e pesca

3 Addestramento delle reti

Le reti scelte per il test sono la VGG16, basata sull'architettura VGG con 16 livelli [Simonyan e Zisserman, 2014] e la ResNet50, basata sull'architettura ResNet con 50 livelli [He *et al*., 2016]. Entrambe sono state implementate in Keras utilizzando Tensorflow come backend e inizializzate con i parametri ottenuti addestrando la rete sul dataset

ImageNet. Tali reti sono state confrontate con la Resnet50 implementata nella libreria Halcon, oltre ad altre due reti con architettura proprietaria di MVTec gmbh presenti nella medesima libreria e denominate 'compact' e 'enhanced'. La libreria consente di operare un fine tuning delle reti su un dataset di immagini fornito dall'utente. Tutte le reti sono state addestrate su un computer con processore Intel® Core™ i7-8700, RAM 16 GB e GPU Nvidia 1080 con 8GB di memoria per 200 epoche, partendo da un learning rate pari a 0.001, implementando una strategia di decadimento del learning rate e con momento pari a 0.9 [Jacobs, 1988]. Il batch size è stato scelto per ottimizzare l'utilizzo della GPU come riportato in tabella 1.

Rete	Batch size	Memoria GPU utilizzata (GB)
VGG16	16	6.5
Resnet50	16	6.5
Halcon Resnet50	16	4.9
Halcon Enhanced	64	4.1
Halcon Compact	64	2.3

Tabella 1: Batch size e utilizzo della GPU

4 Risultati

Tutte le reti utilizzate hanno raggiunto una accuratezza di test del 100% tranne la VGG16 che è arrivata a 99.3%. Il confronto proposto tiene in considerazione la dimensione del classificatore prodotto su disco, legato al numero di parametri, e il tempo di training, legato al numero di operazioni che la rete richiede di eseguire. I risultati sono riportati in tabella 2.

Rete	Dimensioni del classificatore (MB)	Tempo di training su GPU (min)
VGG16	286	32
Resnet50	185	56
Halcon Resnet50	184	62
Halcon Enhanced	81	19
Halcon Compact	6	8

Tabella 2: Dimensioni del classificatore e tempi di training

Sono stati valutati i tempi medi richiesti per classificare una singola immagine, con batch size uguale a 1 (tabella 3).

Rete	Tempo di inferenza su GPU (ms)	Tempo di inferenza su CPU (ms)
VGG16	10.1	166
Resnet50	12.0	67.0
Halcon Resnet50	6.7	33.0
Halcon Enhanced	5.0	12.7
Halcon Compact	1.7	4.4

Tabella 3: Tempi di inferenza CPU i7-8700, GPU Nvidia 1080

Il test sui tempi di inferenza è stato ripetuto su un notebook con processore Intel® Core™ i7-7700, RAM 16 GB e GPU Nvidia Quadro M2200 con 4GB di memoria nelle medesime condizioni. I risultati sono riportati in tabella 4.

Rete	Tempo di inferenza su GPU (ms)	Tempo di inferenza su CPU (ms)
VGG16	31.0	248
Resnet50	27.0	349
Halcon Resnet50	17.0	48.0
Halcon Enhanced	11.7	22.0
Halcon Compact	4.7	6.8

Tabella 4: Tempi di inferenza CPU i7-7700, GPU Nvidia M2200

5 Conclusioni

Le reti convolutive basate su deep learning implementate nella libreria di machine vision Halcon appaiono più efficienti in termini di risorse computazionali e più veloci in inferenza delle reti VGG16 e ResNet50 implementate in Keras utilizzando Tensorflow come backend, pur mantenendo una ottima accuratezza di classificazione.

Riferimenti bibliografici

[He *et al*., 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. *The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770-778, Las Vegas, California, June 2016, IEEE.

[Simonyan e Zisserman, 2014] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations*, *arXiv preprint arXiv:1409.1556*, San Diego (CA), 2015.

[Muresan e Olten, 1985] Horea Muresan and Mihai Oltean. Fruit recognition from images using deep learning. *Acta Universitatis Sapientiae, Informatica* 10(1):26-42, June 2018.

[Jacobs, 1988] Robert A. Jacobs. Increased rates of convergence through learning rate adaptation. *Neural Networks* 1(4): 295-307, 1988.

[Richter e Streitferdt, 2018] J. Richter and D. Streitferdt, Deep Learning Based Fault Correction in 3D Measurements of Printed Circuit Boards, *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, 27-232, Vancouver, BC Canada, November 2018, IEEE.