

# Un sistema di Vision Inspection basato su reti neurali

Ludovico Buffon<sup>1</sup>, Evelina Lamma<sup>1</sup>, Fabrizio Riguzzi<sup>1</sup>, Davide Formenti<sup>2</sup>

<sup>1</sup>Dipartimento di Ingegneria, Via Saragat 1, 44122 Ferrara (FE), Italia  
ludovico.buffon@student.unife.it,  
{evelina.lamma, fabrizio.riguzzi}@unife.it

<sup>2</sup>Bonfiglioli Engineering, Vigarano Pieve (FE), Italia  
d.formenti@bonfiglioliengineering.com

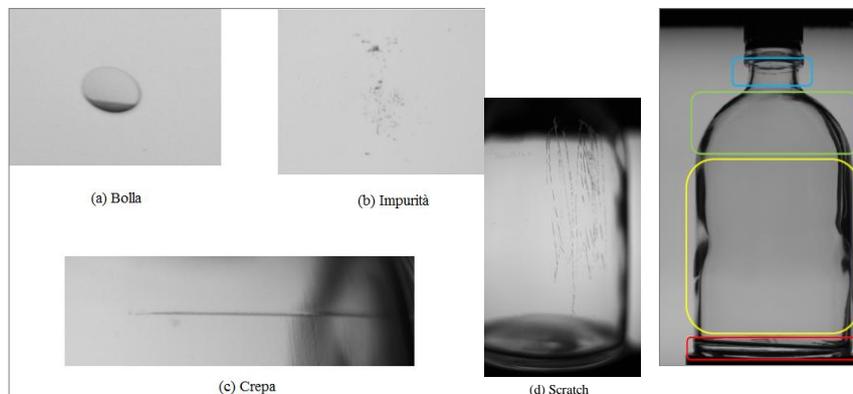
**Abstract.** Il lavoro descrive un sistema di Intelligenza Artificiale per estrarre e riconoscere difetti presenti su flaconi per uso cosmetico e farmaceutico. Il sistema realizzato, nel suo complesso, integra algoritmi di *corner detection* e *clustering* per l'identificazione delle regioni di interesse che costituiscono potenziali difetti, e algoritmi di classificazione automatica basati su reti neurali. Bonfiglioli Engineering s.p.a. ha fornito un database di oltre 20.000 immagini (di flaconi con difetti o integri) a cui fare riferimento, sia per la fase di addestramento sia per quella di *testing* della rete neurale.

**Keywords.** Visione automatica, Classificazione automatica, Reti neurali.

## 1 Introduzione

La visione automatica (*Computer Vision*) ha lo scopo di ricavare informazioni qualitativamente elevate con l'obiettivo di comprendere l'immagine per individuare, ad esempio, qualche *pattern* in essa contenuto. In campo industriale, la visione automatica ricopre ormai un ruolo di primo piano in diverse applicazioni di misura e controllo, grazie alla disponibilità di potenza di calcolo adeguata all'esecuzione di algoritmi di visione e classificazione automatica *real-time* sempre più complessi. I problemi industriali affrontabili mediante visione automatica sono eterogenei [3].

In questo lavoro, descriviamo un sistema software basato su tecniche di Intelligenza Artificiale e di apprendimento automatico per il riconoscimento di *pattern* da immagini ([1,3]), in particolare difetti in flaconi per uso cosmetico e farmaceutico. I principali difetti da ricercare sono bolle d'aria nel vetro, crepe, punti neri, macchie e impurità del materiale, graffi estesi a tutta la superficie (denominati *scratch*). In Figura 1, a sinistra, sono riportati alcuni esempi di difetti. I difetti sono da ricercare su tutto il flacone, in particolare (si veda Fig.1, a destra), nella base, nella parte cilindrica, nella spalla, nel collo del flacone. Il caso di studio è stato proposto da Bonfiglioli Engineering s.p.a. per il miglioramento dei propri prodotti, fornendo anche un database di oltre 20.000 immagini, molto eterogeneo e comprendente immagini di varia grandezza e risoluzione, in cui ogni flacone (difettoso o meno) è fotografato da varie angolazioni.



**Fig. 1.** Esempi di difetti, a dx e centro; Parti del flacone (base, cilindro, spalla, collo), a sx

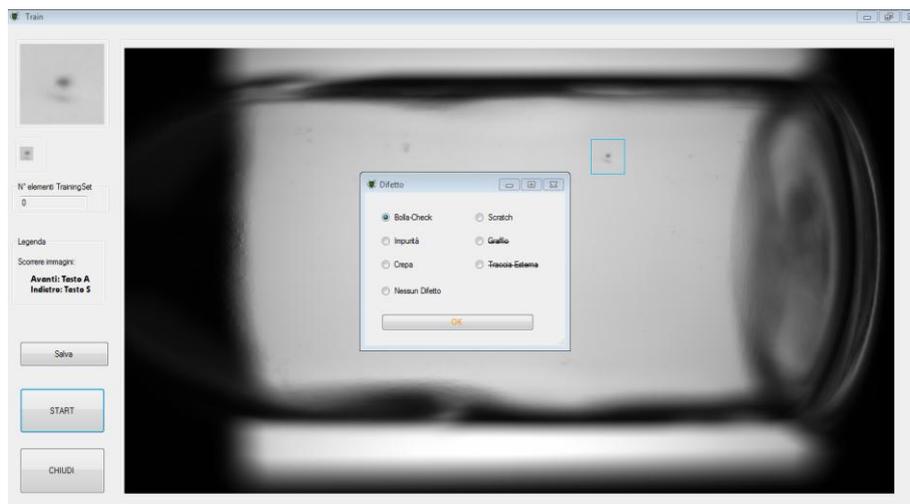
Nell'affrontare il problema, si è preferito un approccio basato su reti neurali rispetto ad altri, perché: (i) è già stato applicato con successo in campi simili (ad esempio, per riconoscimento di volti), (ii) garantisce, inoltre, la capacità della rete di addestrarsi da esempi adattandosi ai *pattern* in ingresso, con un algoritmo efficiente di apprendimento dei pesi della rete (*back-propagation*), e (iii) garantisce la capacità di generalizzazione per trattare *input* mai visti in precedenza. Tuttavia, questo approccio ha presentato inizialmente un ostacolo, perché, per efficienza computazionale, esso è in grado di trattare *pattern* di ingresso di estensione limitata. Per questo motivo, il sistema realizzato integra, nel suo complesso, algoritmi di classificazione automatica basati su reti neurali ([2,6]) e, a monte di questi, applica algoritmi di *corner detection* [7] e *clustering* per identificare nell'immagine (molto estesa in termini di *pixel*) regioni di interesse (*Region Of Interest*, ROI) di area più limitata e le loro caratteristiche, che costituiscono potenziali difetti, successivamente classificate dalla rete neurale addestrata.

## 2 L'architettura software

Il software è stato realizzato in C# con l'ausilio di alcune librerie per il trattamento di immagini *.tiff* e altre per l'apprendimento automatico. In particolare, nello sviluppo, sono state utilizzate la libreria *Accord.net* [4] e la libreria *OpenCvSharp*, porting su piattaforma *.Net* della omonima libreria Intel per visione artificiale. L'interfaccia utente principale comprende ausili per eseguire sia la fase di addestramento della rete, sia la successiva fase di *testing* e di applicazione del classificatore appreso. La topologia della rete neurale (Multi-Layered Perceptron, MLP [8]) ha uno strato di *input*, uno di elaborazione ed uno di *output*. La fase di training avviene sfruttando un algoritmo di *error back-propagation*.

Nella fase di *Training*, si presentano delle immagini all'operatore per allenare la rete neurale al riconoscimento dei *pattern* stabiliti. La fase di addestramento, quindi, è quella che richiede il maggior tempo d'intervento dell'esperto per individuare gli

esempi di *training* più idonei alle circostanze e agli obiettivi dello studio. La scelta delle immagini di *training* deve essere accurata perché, tramite le informazioni in esse contenute, saranno estratte le firme spettrali delle varie classi. Gli *input* della rete sono costituiti da aree estratte dall'esperto dalle fotografie, dette *Region Of Interest* (ROI). Nella fase di *training*, le ROI sono aree di dimensione 30x30 pixel, identificate da parte dell'esperto. Alla rete neurale è data in *input* l'intera ROI effettuando una mappatura uno ad uno fra pixel e neuroni di *input*, insieme alla classe attribuita dall'esperto (si veda Figura 2).



**Fig. 2.** Fase di *training*

Dopo aver opportunamente allenato la rete a riconoscere i difetti proposti, essa può essere utilizzata per una classificazione automatica degli stessi. Il sistema si compone di due passi: la prima di estrazione automatica delle caratteristiche, la seconda di classificazione automatica. Lo schema complessivo è riportato in Figura 3.

La fase di *feature extraction* si divide, a sua volta, in due passi in cascata. Nel primo, si analizza l'immagine con l'algoritmo di *corner detection* FAST (Features from Accelerated Segment Test [7]) per selezionarne i punti caratteristici (o *key point*), salvati poi in un vettore di punti in coordinate cartesiane. Successivamente questi punti sono raggruppati in *cluster* tramite l'algoritmo DBScan (Density-Based Spatial Clustering of Applications with Noise [5]). Infatti i punti che l'algoritmo FAST individua si addensano in prossimità dei difetti presenti nell'immagine e quindi i *cluster*, opportunamente filtrati per mantenere quelli con più punti al loro interno, individuano regioni spaziali dove c'è più probabilità di avere un difetto. Dopo aver selezionato i *cluster* di interesse, si estrae una regione di 30x30 pixel (Region Of Interest, ROI), passata poi alla rete neurale (MLP, in Figura 3) per essere classificata. La classe individuata è quella per cui la probabilità calcolata dalla rete è massima [9].

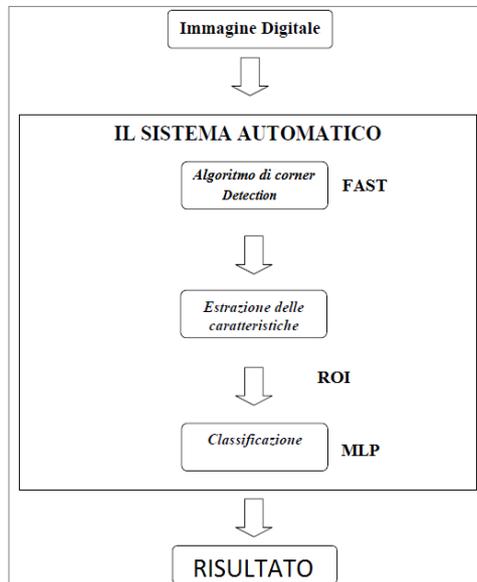


Fig. 3. Schema del sistema di classificazione automatica

### 3 Sperimentazione e risultati

Il database messo a disposizione contava più di 20.000 immagini, comprensivo sia di immagini di flaconi con difetti sia non difettosi, fotografati ciascuno da quattro diverse angolazioni. La fase di addestramento e la successiva di test sono state effettuate su un numero ridotto di immagini, circa 12.000, partizionate in due insiemi, rispettivamente per il *Training* (7000 immagini) e il *Test* (5000 immagini) della rete.

Dal punto di vista hardware la fase di *Test*, che è quella che richiede la maggior parte delle risorse della macchina su cui è eseguito il programma, occupa al massimo un quantitativo di memoria pari a 250 Mb ed esegue l'elaborazione e la classificazione di un'immagine in un tempo non superiore ai 2 secondi.

Nella fase di *Test* si è valutata inizialmente la bontà del sistema proposto analizzando le immagini con sole due possibili classi di appartenenza: Difetto/Non-Difetto. In questa prima sperimentazione svolta, si è valutata la capacità del sistema sviluppato addestrando la rete con 200 esempi selezionati dal *dataset* di 7.000 immagini. La scelta di usare solo 200 immagini nel *training* è dovuto al fatto che questa fase è risultata molto lunga e dispendiosa in termini di tempo uomo. Il database di *test* conteneva invece le restanti circa 5.000 immagini. In questa fase circa il 97% delle immagini (4850 immagini su 5000) è stato classificato correttamente.

In seguito, si è aumentato il numero di classi per avere una classificazione più precisa. Nell'attuale sviluppo del sistema, il numero di classi che la rete neurale riesce a discriminare è pari a cinque: Non-Difetto, Bolla, Crepa, Impurità, più la classe denominata Scratch. Con l'aumento del numero di classi, nella seconda sperimenta-

zione condotta, per mantenere una buona classificazione, si è dovuto aumentare sia il numero di esempi forniti alla rete (portato a 320 esempi) sia il numero di cicli di apprendimento, e diminuire il *learning rate* per evitare ampie oscillazioni dell'errore medio sul *training set*. I tempi di apprendimento si sono allungati fino a qualche ora con 320 esempi. È stato poi preso in considerazione per il *test* solo un sottoinsieme di 500 immagini prelevate a caso dal database di 5.000 totali per un ammontare di circa 1.000 difetti estratti e analizzati. I risultati evidenziano che è rimasta circa invariata la capacità di *detection* delle problematiche presenti dovuto al fatto che FAST è un algoritmo con un'ottima stabilità e ripetibilità nell'individuazione dei punti interessanti. La classificazione più accurata ha evidenziato un errore del 7-8% nella corretta classificazione della regione estratta. All'interno delle 5 classi la probabilità maggiore di errore è data dalla discriminazione tra Crepa e Impurità in regioni con poco contenuto informativo, mentre la classificazione come Bollia ha dato il 98% di accuratezza. Un esempio di elaborazione è mostrato in Figura 4, con l'immagine di ingresso a sinistra, e i *cluster* interessanti e classificati evidenziati direttamente sull'immagine a destra per un immediato riscontro visivo.

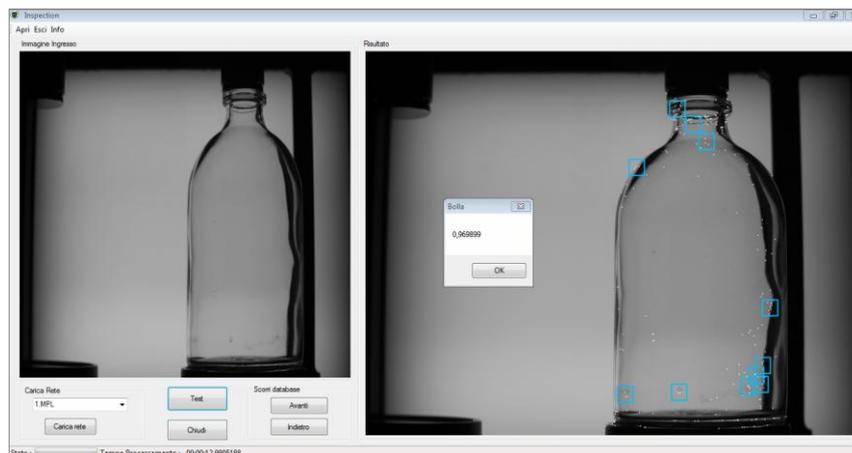


Fig. 4. Classificazione dei *cluster* interessanti

## 4 Conclusioni e sviluppi futuri

In questo lavoro abbiamo integrato reti neurali e algoritmi di *corner detection* e *clustering* automatico per l'estrazione delle caratteristiche visuali da immagini in un sistema software per l'identificazione di difetti in flaconi ad uso cosmetico e farmaceutico, prodotti dai macchinari sviluppati da Bonfiglioli Engineering s.p.a. Il sistema realizzato è, per ora, funzionante *off-line* rispetto alla linea di produzione dell'azienda. Il sistema, dopo la fase di addestramento della rete neurale, ha evidenziato un'ottima capacità di individuare i difetti presenti all'interno delle immagini e una buona capacità di classificare il *pattern* estratto.

Il processo di ricerca dei parametri ottimali per l'algoritmo di classificazione automatica realizzato dalla rete neurale è stato lo scoglio maggiore del progetto. Sono stati provati diversi valori di inizializzazione in modo tale da avvicinarsi ad un buon comportamento della rete, e ottenendo buoni risultati sperimentali.

Grazie alla modularità del software realizzato, si possono facilmente inserire ulteriori difetti, rispetto a quelli già considerati, già presenti nel database di immagini fornito dall'azienda, come *TracceSottili*, *Graffi*, e *TracceEsterne*.

Ulteriori sviluppi futuri riguardano l'integrazione di nuovi classificatori come macchine SVM, Hidden Markov Model, AdaBoost, Reti di Kohonen (SOM), già previsti nella struttura del codice e parzialmente considerati in un lavoro parallelo a questo come approcci alternativi.

## **Ringraziamenti**

Il lavoro è stato sviluppato nell'ambito del progetto dal titolo *Elaborazione delle Immagini e Visione Artificiale per la Classificazione di Immagini in ambito Industriale*, finanziato con il contributo della Camera di Commercio, Industria, Artigianato e Agricoltura di Ferrara, Bando CCIAA-2010.

## **Riferimenti bibliografici**

1. Bishop, C.M., "Pattern recognition and machine learning". Springer, 2006.
2. Bishop, C. M., "Neural Networks for Pattern Recognition". Oxford University Press, 1995.
3. Cucchiara R., Mello P., Piccardi M., Riguzzi F., "An application of machine learning and statistics to defect detection", *Intelligent Data Analysis*, 5(2):151-164, March/April 2001.
4. <http://accord-net.origo.ethz.ch>
5. <http://it.wikipedia.org/wiki/Dbscan>
6. Leoneds, C.T, (editor), "Image processing and pattern recognition - Neural network system techniques and applications", Academic Press, 1998.
7. Rosten, E., Drummond, T., "Machine learning for high-speed corner detection", *Proc. European Conference on Computer Vision*, vol. 1, pp. 430-443, 2006.
8. Rosenblatt, F., "The perceptron: A probabilistic model for information storage and organization in the brain". *Cornell Aeronautical Laboratory, Psychological Review*, 65:386, 1958.
9. Vapnik, V.N., "The Nature of Statistical Learning Theory", second edition, Springer, 1999.