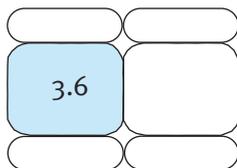




PATTERN RECOGNITION TECNICHE E APPLICAZIONI

Alessio Malizia



Molti bambini, già in tenera età, sono in grado di riconoscere chiaramente lettere e numeri, scritti in corsivo o a macchina. Possiamo pensare che questa attività di riconoscimento, nota come *Pattern Recognition*, sia semplice finché non tentiamo di automatizzarla, ovvero di far riconoscere automaticamente simboli e caratteri ad un computer. L'obiettivo di questo articolo è di sintetizzare i principali metodi di *Pattern Recognition* e presentare le linee della ricerca nel settore.

1. INTRODUZIONE

Il riconoscimento automatico di oggetti (*pattern*) e la loro descrizione, classificazione e raggruppamento (*clustering*) sono argomenti importanti in una grossa varietà di problemi sia nell'area ingegneristica che scientifica, quali: la biologia, la psicologia, la medicina, il marketing, la visione artificiale, l'intelligenza artificiale fino alla fusione di dati (meglio conosciuta come *data fusion*).

La domanda che sorge spontanea nel descrivere la *pattern recognition*, è: cos'è esattamente un *pattern*?

Watanabe [18] descrive un *pattern* come "l'opposto del caos; ovvero un'entità, generalmente definita a cui si può dare un nome". Per esempio, un *pattern* può essere l'immagine di una impronta digitale, una parola scritta in corsivo, l'immagine di un volto, o un segnale acustico.

Dato un *pattern*, il suo riconoscimento (detto anche classificazione) può essere principalmente effettuato in due modi [18]:

1. supervisionato, in cui il *pattern* in input viene identificato come membro di una clas-

se tra quelle predefinite, ovvero viene classificato in maniera supervisionata avendo l'utente o il progettista definito a priori le classi di interesse;

2. non supervisionato, in cui il *pattern* in input viene assegnato ad una classe sconosciuta a priori, ovvero i *pattern* vengono raggruppati nei cosiddetti *cluster* (tecniche di *clustering*).

Il problema della *pattern recognition* quindi viene posto nella forma di classificazione o identificazione delle categorie di appartenenza, dove le classi o categorie possono essere sia definite dal progettista del sistema (nei metodi supervisionati), sia apprese dalle similarità tra i *pattern* (nei metodi non supervisionati).

I sistemi di *pattern recognition* creano, chiaramente, nuovi scenari applicativi sia a livello accademico che industriale. Si pensi, per esempio, ai vantaggi per un processo industriale che, impiegando il riconoscimento automatico di oggetti, può beneficiare di un controllo qualità automatizzato. Sempre nell'ambito industriale, l'impiego di algoritmi di

riconoscimento automatico, porta ad arricchire le funzionalità dei robot impiegati, come ad esempio la possibilità di muoversi in determinati ambienti riconoscendo automaticamente gli ostacoli. Altre applicazioni di interesse, soprattutto ai giorni nostri, sono quelle di video sorveglianza automatizzata. Pensiamo, ad esempio, ad una stazione ferroviaria o ad un aeroporto ed a come, vista la mole di persone in transito, un riconoscimento automatico di oggetti incustoditi possa essere cruciale per le funzionalità di sorveglianza. Il riconoscimento di volti, è un'altra area di applicazione che consente di aumentare notevolmente la sicurezza sia in ambito pubblico che aziendale. Un ulteriore scenario, mol-

to interessante, riguarda la cosiddetta "Realtà Aumentata" dove informazioni reali percepite da utenti umani vengono arricchite da informazioni sull'ambiente estratte automaticamente da un sistema automatizzato. Un esempio tipico è quello militare con la possibilità di avere visori che includono tracciamento automatico dei soggetti individuati e stima della distanza dall'osservatore. Già da questi primi esempi è chiaro come l'uso di tecniche di riconoscimento automatico possa notevolmente migliorare le attuali applicazioni tecnologiche in diversi campi, dal settore scientifico a quello sociale ed industriale. La tabella 1 mostra, in sintesi i principali domini applicativi dei sistemi di riconoscimento

Dominio	Applicazioni	Pattern	Classi
Analisi di documenti	Ricerca su Internet e Semantic Web	Documenti testuali / corpus di documenti digitali o digitalizzati	Categorie semantiche (esempio, affari, sport, scienza ecc.)
Analisi di immagini di documenti	Strumenti di supporto per i diversamente abili, digitalizzazione automatica di documenti	Immagini di documenti (esempio, acquisite da scanner)	Riconoscimento dei caratteri (OCR), parole, regioni di testo
Automazione Industriale	Verifica delle schede di circuiti, verifica del software	Immagini all'infrarosso o spettroscopiche, e moduli software	Rilevamento di componenti hardware e software difettose
Data Mining	Ricerca di pattern significativi in insiemi di dati/documenti	Punti nello spazio multidimensionale / vettori multidimensionali	Gruppi omogenei e ben separati (cluster)
Bioinformatica	Analisi sequenze DNA	Sequenze DNA/Proteine	Tipi di geni/proteine
Information Retrieval su basi dati multimediali	Ricerca su internet o in reti aziendali di immagini ed audio/video	VideoClip ed immagini	Generi video, tipi di immagini o tipologia di inquadrature (ed. nello sport: goal, rigore, fuorigioco ecc.)
Riconoscimento BioMetrico	Identificazione personale attraverso rilevazione dati biometrici	Immagini del volto, dell'iride, dell'impronta digitale	Gestione controllata accessi
Riconoscimento del parlato	Dettati automatici, inserimento dati per persone diversamente abili, interfaccia audio per l'uso di telefonia	Onde sonore della voce	Parole pronunciate, suoni modulati
Sensor Data Fusion	Riconoscimento della persona e della posizione geografica attraverso diversi sensori	Informazioni provenienti da sensori (esempio, accesso mediante RFID - identificazione a radio frequenza)	Posizione, risorse disponibili, contesto dell'utente

TABELLA 1

Applicazioni della pattern recognition

automatico evidenziando anche le classi di oggetti/entità coinvolte nei suddetti processi. Le principali aree di applicazione della Pattern Recognition sono:

- Automazione dei processi industriali.
 - Riconoscimento di oggetti, ispezioni visuali, controlli di qualità.
 - Navigazione e coordinazione di robot.
- Sorveglianza e tracking.
 - Traffico, sicurezza.
- Interazione uomo-computer.
 - Rilevamento e riconoscimento di volti.
 - Assistenza ai disabili.
- Realtà virtuale e visualizzazione.
 - Ricostruzione di scene e modelli 3D.
 - Fotometria.
- Applicazioni spaziali e militari.

Tra i campi più innovativi nell'applicazione del riconoscimento di *pattern*, c'è ad esempio l'*affective computing*, identificato da Picard [18], che introduce la possibilità da parte del sistema di riconoscere e esprimere emozioni, al fine di rispondere in maniera intelligente alle espressioni umane.

In genere, un sistema di *pattern recognition*, riceve in ingresso la descrizione di un oggetto, ovvero un insieme di misure che lo caratterizzano (*feature*) e sulla base di queste "decide" a quale classe l'oggetto appartiene. La decisione della classe di appartenenza ha un costo associato all'assegnazione dell'oggetto alla classe sbagliata e l'obiettivo con cui realizzare un buon sistema di *pattern recognition* è quello di minimizzare il costo di classificazione.

Comunque, la caratteristica comune a tutti questi sistemi e domini applicativi è che le misure caratteristiche (*feature*) non sono suggerite o identificate dagli esperti di dominio, ma vengono estratte ed ottimizzate da procedure che elaborano dati; questo avviene perché in genere abbiamo migliaia di misure possibili anche per un singolo *pattern*.

La crescente disponibilità di risorse per il calcolo automatico non solo consente di elaborare contemporaneamente grosse moli di dati, ma permette anche l'analisi e la classificazione di grandi insiemi di *pattern*. Allo stesso tempo, la domanda di applicazioni di *pattern recognition* cresce continuamente grazie alla disponibilità di grosse basi di dati e del calo dei costi dei supporti di memorizzazione. Tut-

tavia, in molte delle applicazioni di *pattern recognition* risulta evidente che non esiste un approccio "ottimale" e "generale" alla classificazione e che la scelta di integrare diversi metodi e approcci è la migliore da perseguire. La progettazione di un sistema di *pattern recognition* richiede essenzialmente di affrontare i seguenti aspetti:

1. Acquisizione e pre-elaborazione (e normalizzazione) dei dati.
2. Rappresentazione e classificazione dei dati o *pattern*.
3. Decisione e classificazione.

Il dominio di applicazione, in pratica, influenza enormemente la scelta dei sensori (nel senso generale del termine), delle tecniche di pre-elaborazione e di normalizzazione dei dati, della rappresentazione degli stessi e del modello decisionale di classificazione.

È noto che per i problemi di riconoscimento ben definiti e quindi condizionati opportunamente (esempio: piccole variazioni nei valori appartenenti alla stessa classe e grosse variazioni in quelli appartenenti a classi diverse) esistono soluzioni efficienti e buone strategie decisionali di classificazione. Per esempio, usare approcci orientati all'apprendimento, basandosi su un insieme di dati per l'addestramento automatico del sistema (*training set*) consente la necessaria flessibilità del sistema, che addestrato con dati diversi, può reagire a situazioni diverse con un margine di errore controllabile.

I quattro principali approcci alla *pattern recognition* [9] sono:

1. approcci basati su modello (*Template Matching*);
2. approcci con classificazione statistica (*Statistical Classification*);
3. approcci sintattici o strutturali (*Syntactic or Structural Matching*);
4. approcci basati su reti neurali (*Neural Networks*).

Questi approcci non sono necessariamente separati ed indipendenti, infatti a volte lo stesso metodo può essere usato nei diversi approcci con diverse interpretazioni. Esistono anche tentativi di progettazione e realizzazione di sistemi ibridi [5] che introducono e integrano più modelli all'interno dello stesso sistema. Nei paragrafi successivi verranno descritti e analizzati i quattro approcci menzionati al

Approccio	Rappresentazione	Riconoscimento	Criterio di Classificazione
Basato su Modello (<i>Template Matching</i>)	Dati campionati, pixel, curve introduzione di distanze	Correlazione statistica,	Basato su stima dell'Errore di Classificazione
Classificazione Statistica (<i>Statistical Classification</i>)	Misure o Caratteristiche (Feature)	Funzioni discriminanti	Basato su stima dell'Errore di Classificazione
Sintattico o Strutturale (<i>Syntactic or Structural Recognition</i>)	Primitive	Regole, Grammatiche	Basato su stima dell'Errore di Accettazione
Reti Neurali (<i>Neural Network</i>)	Dati campionati, pixel, Caratteristiche (Feature)	Funzioni sinaptiche e di rete	Basato su stima dell'Errore quadratico medio

TABELLA 2

Approcci e modelli per i sistemi di pattern recognition

fine di evidenziarne non solo le caratteristiche ma valutarne i vantaggi e gli svantaggi alla luce dei diversi domini applicativi. Nella tabella 2 vengono presentati i diversi approcci con una descrizione sintetica delle loro caratteristiche.

2. SISTEMI DI PATTERN RECOGNITION BASATI SUL MODELLO

Uno dei primi e più semplici degli approcci alla pattern recognition è quello basato sul modello, il cosiddetto *template matching*. I sistemi di *pattern recognition* basati sul modello utilizzano la cosiddetta tecnica del "matching"; il *matching* è definito come l'operazione che viene usata per determinare la similarità tra due entità dello stesso tipo: punti, curve o forme ad esempio. Nell'approccio basato su modello, quest'ultimo (il modello) è tipicamente un prototipo del *pattern* od oggetto da riconoscere e classificare, tipicamente una forma bidimensionale. L'oggetto da riconoscere (il *pattern*) viene confrontato con tutti i modelli memorizzati nel sistema considerando i cambiamenti di scala e le possibili invarianze per rotazione o traslazione. Naturalmente ci saranno casi in cui sarà utile l'invarianza di scala, cioè ci interessa riconoscere l'oggetto semplicemente dalla forma, mentre in altri casi sarà necessario distinguere non solo rispetto alla forma ma anche rispetto alla dimensione. Nella figura 1 è riportato un esempio di sistema basato su modello [1] dove si mostra un *pattern*, l'aereo in input (A), e l'estrazione della forma (B). In

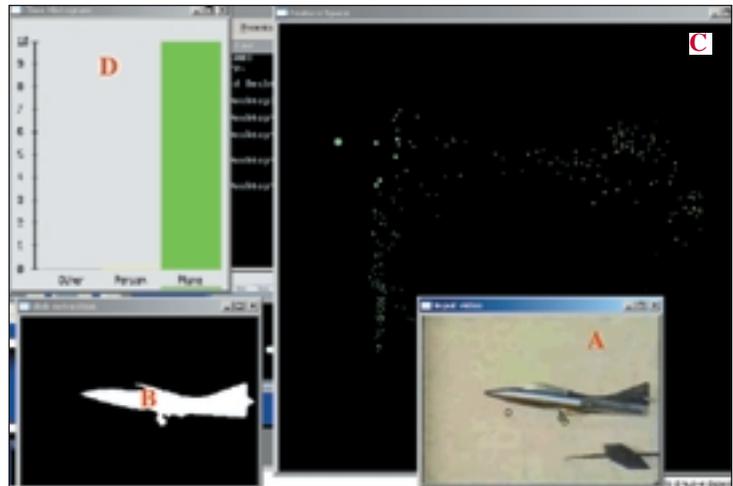


FIGURA 1

Esempio di sistema di pattern recognition basato su modello: A pattern in input (aereo), B estrazione forma, C base dati di addestramento (training set), D istogramma di classificazione

figura è presente anche una finestra sullo sfondo (C) con dei punti rappresentanti i valori delle caratteristiche per i modelli di aereo presenti nella base dati; i punti di dimensione più grande rappresentano i modelli che sono più simili all'aereo in input (risultato della fase di *matching*). L'istogramma di colore verde (D), mostra come l'oggetto sia classificato "aereo" dal sistema, perché la maggior parte (tutti in questo caso) dei punti della base dati di oggetti nel suo intorno erano stati classificati (in fase di addestramento) come aerei (colore verde).

La misura di similarità utilizzata nei sistemi basati su modello, è solitamente una correlazione statistica e può essere ottimizzata apprendendo i parametri dal *training set*,

ovvero da un insieme di esempi di oggetti forniti in input al sistema. I sistemi basati su modello sono computazionalmente inefficienti, richiedono molte risorse di calcolo per confrontare il modello del pattern in input con tutti gli altri presenti nella base dati, ma grazie alla continua crescita della potenza di calcolo questo approccio viene sempre più utilizzato soprattutto in ambito industriale. Per come abbiamo definito il modello fino ad ora si dovrebbe parlare di sistemi basati su modelli rigidi, ovvero sistemi dove il modello è descritto con parametri ed insiemi di valori fissati. Questo tipo di sistemi tende a non essere efficiente in presenza di distorsioni dovute all'acquisizione dell'oggetto (per esempio: effetti di errori sulle immagini), cambiamento dei punti di vista, o grosse variazioni di valori delle caratteristiche all'interno della stessa classe (*inter-class variance*). In questi ultimi casi si tende ad utilizzare tecniche di modelli deformabili o elastici [8] al fine di consentire un confronto di similarità (*matching*) tra modelli le cui deformazioni non possono essere rappresentate esplicitamente a priori. Per intenderci, questi sistemi sono utili quando non possiamo modellare a priori tutte le possibili deformazioni che un oggetto possa avere, ed allora si agisce creando un modello dell'oggetto e studiando quali agenti esterni lo possono deformare (modellando quin-

di le possibili deformazioni) a volte simulando anche forze fisiche.

3. SISTEMI DI PATTERN RECOGNITION CON CLASSIFICAZIONE STATISTICA

Nell'approccio statistico ogni pattern è rappresentato in termini di un vettore di lunghezza d , dove d indica il numero di misure caratteristiche o feature dello stesso e viene rappresentato come un punto nello spazio d -dimensionale delle caratteristiche (*feature o vector space*). L'obiettivo, quindi, è quello di selezionare un insieme di feature che consenta a vettori appartenenti a categorie differenti di occupare spazi compatti in regioni disgiunte dello spazio d -dimensionale delle feature. Nell'esempio di figura 2 [3] è rappresentato uno spazio con due caratteristiche, la dimensione e la luminosità; queste due caratteristiche sono usate in questo esempio per distinguere immagini di pesci, più esattamente tra due classi di pesci: i salmoni e le spigole. Si può notare come i valori per ogni pesce nello spazio delle caratteristiche si distribuiscono in due regioni, legate alla dimensione ed alla luminosità della livrea dei pesci.

L'efficacia della rappresentazione spaziale delle feature (*feature set*) è rappresentata da come i pattern delle diverse classi sono ben separati nello spazio. Dato un insieme di addestramento (esempi di pattern) per ogni classe, l'obiettivo è quello di stabilire delle regioni di decisione nello spazio delle feature che consenta di separare i pattern appartenenti a classi diverse; nell'esempio di figura 2 la regione di decisione è data dalla retta che separa (con un piccolo margine di errore) le due classi di pesci.

Nell'approccio statistico, le regioni di decisione sono determinate dalle distribuzioni di probabilità per ogni pattern di appartenere ad una delle classi; le distribuzioni vanno quindi specificate (anche parametricamente) oppure apprese mediante addestramento del sistema (*machine learning*) [2, 4].

Uno degli approcci più utilizzati è quello dell'analisi dei discriminanti per la classificazione:

□ prima di tutto si sceglie, sulla base dell'esperienza una forma parametrica per la re-

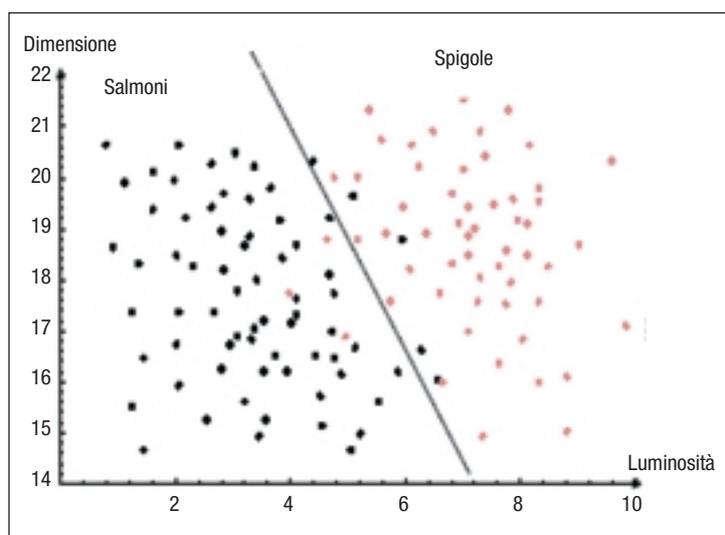


FIGURA 2
Spazio bidimensionale delle caratteristiche per due classi di pesci, i salmoni e le spigole

gione di decisione, per esempio lineare (figura 2) o quadratica;

□ successivamente si sceglie, sulla base dell'insieme di addestramento, la migliore regione di decisione ai fini della classificazione.

In genere si usano metodi statistici per la stima degli errori per determinare le regioni di decisione. Il motivo di questa scelta si basa sul fatto che la migliore regione di decisione è quella che minimizza l'errore nell'assegnare il pattern alla classe sbagliata.

Uno degli altri principali approcci alla costruzione delle regioni di decisione, l'approccio diretto, nasce dalla cosiddetta filosofia di Vapnik [17]. Questa si basa sull'idea che se si possiede un insieme ridotto di informazioni sul problema che si sta affrontando, è meglio cercare di risolvere il problema direttamente, senza affrontare generalizzazioni dello stesso come passo intermedio. È possibile, infatti, che in questo caso si posseggano informazioni sufficienti solo per una *soluzione diretta* dello stesso, ma che le stesse siano insufficienti per una soluzione più generale.

4. SISTEMI DI PATTERN RECOGNITION CON APPROCCI SINTATTICI O STRUTTURALI

In molti problemi di *pattern recognition* che coinvolgono oggetti complessi, una soluzione può essere quella di considerare una prospettiva gerarchica, dove i pattern vengono visti come composti da subpattern più semplici [6, 14].

I *pattern elementari* o *subpattern* sono chiamati *primitivi* e i pattern complessi da riconoscere sono rappresentati sotto forma di insiemi e relazioni tra primitivi. Nei sistemi di *pattern recognition* sintattici si crea un'analogia formale fra la struttura dei pattern complessi e la sintassi di un linguaggio. I pattern vengono visti come frasi di un linguaggio, mentre le primitive vengono rappresentate come l'alfabeto del linguaggio, e le frasi quindi sono generate a partire da una grammatica per questo linguaggio. Quindi un pattern viene riconosciuto o classificato se la rispettiva frase appartiene al linguaggio specificato per quella classe ed è quindi accettata dalla grammatica specificata. Quindi un grosso insieme di pattern

complessi può essere descritto da un semplice insieme di primitive e regole grammaticali. La grammatica per ogni classe di pattern può essere inferita o appresa dall'insieme di pattern di addestramento o di esempio per ognuna delle classi di interesse.

La *pattern recognition* strutturale (o sintattica) è molto interessante perché, oltre alla classificazione dei pattern, fornisce anche una descrizione di come essi siano costruiti a partire dalle primitive, da qui l'uso della parola strutturale.

Questo approccio è molto usato in situazioni in cui la struttura dei pattern può essere catturata da insiemi di regole, come per le forme d'onda degli elettrocardiogrammi, le *texture* o trame (rappresentazioni fotorealistiche applicate ad un modello sintetico d'immagine) e l'analisi dei contorni [6].

L'implementazione di un approccio sintattico alla *pattern recognition* è in generale piuttosto complessa e questo è dovuto per esempio al problema della **segmentazione delle immagini** al fine di estrarre le primitive ed all'inferenza per estrarre la grammatica dall'insieme di pattern di addestramento. Negli anni ottanta, Fu [6] introdusse la nozione di grammatica con attributi che unifica i concetti di *pattern recognition* sintattico con quello basato su classificazione sta-

Segmentazione di immagini

La segmentazione è un partizionamento di immagini digitali in più regioni (insiemi di pixel) in base ad un certo criterio stabilito. L'obiettivo della segmentazione è tipicamente quello di individuare gli oggetti di interesse che possano essere estratti dall'immagine originale. Tipicamente la maggior parte degli algoritmi di segmentazione sono troppo semplici per risolvere completamente questo problema; la segmentazione può essere vista anche come un tipico problema di visione artificiale (o computer vision).

Un semplice ma tipico esempio può essere quello di usare una soglia fissa per immagini a toni di grigio. Fissato un valore di grigio soglia t : ogni pixel p è assegnato ad una tra le due classi (o regioni), P_0 o P_1 , a seconda se $L(p)$, che è il valore di grigio (tipicamente tra 0 corrispondente al nero, 255 al bianco, e la gamma di valori intermedi alle tonalità di grigio) è tale che $L(p) < t$ o $L(p) \geq t$.



Immagine originale



Immagine dopo l'operazione di soglia

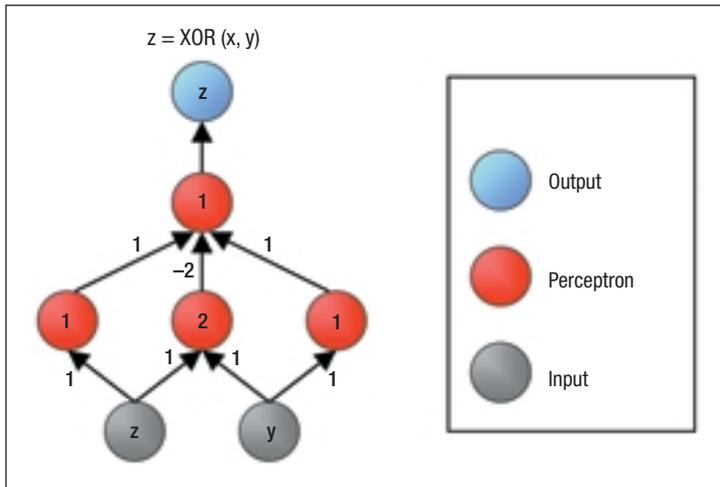


FIGURA 3
Una rete neurale del tipo multi-layer perceptron a due livelli in grado di calcolare la funzione XOR di due variabili

tistica. In effetti, un problema dell'approccio sintattico è che può generare una esplosione di combinazioni da investigare per inferire dalle primitive e dalle loro relazioni la grammatica conseguente; per cui necessita di grandi insiemi di pattern per l'apprendimento e quindi grosse risorse di elaborazione dati [15].

5. SISTEMI DI PATTERN RECOGNITION BASATI SU RETI NEURALI

Le reti neurali possono essere considerate come un sistema computazionale fortemente parallelo costituito da un grande numero di processori estremamente semplici ma intensamente interconnessi fra loro.

I modelli basati su reti neurali cercano di utilizzare principi di organizzazione dei dati, come: addestramento, generalizzazione, tolleranza ai guasti e distribuzione dei dati, in una rete di grafi pesati; i nodi sono rappresentati da neuroni artificiali e gli archi diretti (pesati) rappresentano le connessioni tra l'output di un livello di neuroni e l'input del successivo livello, come è mostrato in figura 3 [19].

La caratteristica principale delle reti neurali è la loro capacità di addestrarsi, mediante procedure iterative, e gestire relazioni non lineari fra l'input e l'output, adattandosi ai pattern di esempio.

Le reti neurali vengono divise in famiglie a seconda della loro struttura e del tipo di connessioni tra i neuroni artificiali; la famiglia di reti neurali più utilizzata per i sistemi di *pattern recognition*, e quindi per la classificazione [10], è quella basata sulle cosiddette *feed-forward network*, che includono tra gli altri i *Multi-Layer Perceptron* (MLP) e le *Radial-Basis Functions* (RBF). Queste reti sono organizzate per livelli e hanno connessioni unidirezionali tra i neuroni appartenenti ai differenti livelli.

Un altro insieme, molto usato, di reti neurali è costituito dalle *Self-Organizing Map* (SOM), o anche reti di Kohonen [11], che sono principalmente usate per il raggruppamento automatico di pattern, chiamato *data cluster*.

Il processo di addestramento nelle reti neurali, consiste nell'aggiornare i pesi di connessione sugli archi dei neuroni artificiali al fine di migliorare la precisione dei risultati della rete a seconda del compito assegnato, di classificazione o di raggruppamento (*clustering*).

Il crescente utilizzo dei modelli per la *pattern recognition* basati su reti neurali è dovuto principalmente alla caratteristica per cui sembra che le reti neurali siano indipendenti dal dominio applicativo del sistema di *pattern recognition*, in contrasto con quanto invece avviene per i sistemi basati su modello o con approccio sintattico/strutturale. In più, per le reti neurali sono disponibili algoritmi efficienti per l'apprendimento da insiemi di esempio dei pattern da classificare.

In definitiva le reti neurali forniscono un insieme di algoritmi non lineari per l'estrazione delle feature dai pattern (mediante l'utilizzo del livello nascosto o *hidden layer*) e per la classificazione dei pattern (per esempio i *multi-layer perceptron*).

In più, gli algoritmi di estrazione delle feature e di classificazione dei pattern possono essere direttamente applicati (matematicamente) su realizzazioni hardware delle reti neurali ai fini di incrementarne l'efficienza.

Diversamente da quello che sembra, molti dei più noti modelli di reti neurali sono implicitamente equivalenti o molto simili ai metodi di *pattern recognition* basati su modelli statistici, come si può vedere dallo schema di un generico modello statistico

presentato in figura 4. Ma indipendentemente dalle similitudini, le reti neurali offrono molti vantaggi ai progettisti di sistemi di *pattern recognition*, quali: un approccio unificato alla selezione ed estrazione delle feature, ed una strategia generale e flessibile per trovare soluzioni ai problemi di classificazione in cui le regioni di decisione siano non lineari.

6. SISTEMI DI RICONOSCIMENTO AUTOMATICO PER BASI DATI MULTIMEDIALI

Approfondiremo, ora, i sistemi di riconoscimento basati su immagini [13] (che utilizzano, in genere, un mix delle tecniche presentate), visto che al giorno d'oggi viene prodotta una sempre maggiore quantità di documenti multimediali (immagini, filmati ecc.), per lo più memorizzati in forma digitale. L'enorme diffusione dei dati multimediali in rete negli ultimi anni ha portato alla crescita di basi dati sia di immagini che di sequenze video. Tuttavia, l'uso efficace di immagini in vari domini applicativi è ostacolato dalla difficoltà di classificare e gestire questi dati in modo efficiente. Sebbene la "navigazione" di sequenze di immagini sia un tema importante in molte applicazioni nelle quali si richiede all'utente di selezionare poche sequenze tra molte, o quando deve trovare un punto particolare all'interno di una sequenza, gli strumenti a disposizione per eseguire tali attività rimangono piuttosto primitivi.

Per progettare un sistema per l'interrogazione di basi dati di immagini è necessario stabilire:

1. quale rappresentazione adottare per descrivere il contenuto delle immagini,
2. in quale modo organizzare le informazioni estratte per poi memorizzarle,
3. che tipo di strategie utilizzare per il recupero e la visualizzazione delle immagini memorizzate.

Nella selezione del tipo di rappresentazione delle immagini, si deve tentare di trovare un insieme di caratteristiche che descrivano nella maniera più completa possibile le immagini presenti nella base dati. Molte delle tecniche adottate per rappresentare il contenuto delle immagini si basano sulla conoscenza, cioè utilizzano informazioni note a priori sul

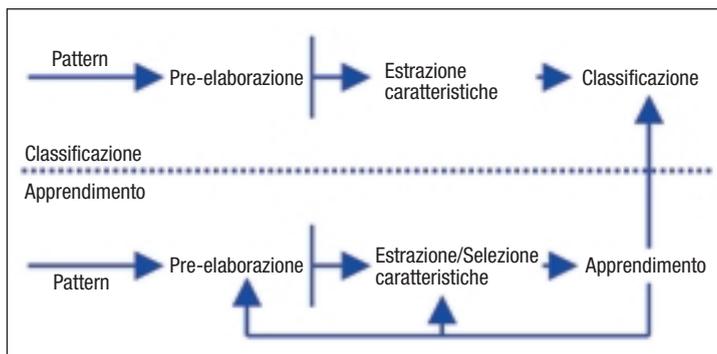


FIGURA 4

Schema generale di un sistema di pattern recognition basato su classificazione statistica. Come si può vedere dalle fasi rappresentate in figura, lo schema descrive anche in generale i sistemi basati su reti neurali

dominio specifico a cui è rivolta l'applicazione, non facilmente utilizzabili per il recupero di immagini in basi dati generiche. In genere gli oggetti possono essere caratterizzati mediante: la loro forma (contorno, baricentro ecc.), le regioni omogenee (segmentazione), o caratteristiche geometriche di vario tipo (esempio, scheletro – tecnica che consiste nel rimuovere ripetutamente i pixel del contorno fino a ridurre l'oggetto a uno "scheletro" largo un pixel) [12].

La memorizzazione del contenuto delle immagini è organizzata mediante indici per rendere più efficiente la successiva fase di recupero. La capacità di recupero delle informazioni in una base dati multimediale è strettamente legata anche al linguaggio per le interrogazioni fornito con essa. Esistono svariate forme di interrogazione su immagini, come ad esempio le interrogazioni tramite identificatori, le interrogazioni testuali, le interrogazioni basate su regioni (viene scelta una regione dell'immagine e tutte quelle che la intersecano vengono selezionate) ecc..

Infine per ottenere i risultati (le immagini), gli indici delle immagini memorizzate sono confrontati con i parametri estratti dalla descrizione dell'immagine fornita in input dall'utente. La rappresentazione visuale dei risultati, quindi, consente all'utente di valutare i risultati ottenuti e di variare gli intervalli entro i quali devono essere compresi i parametri legati ai modelli degli oggetti recuperati, in modo da ottenere risultati più o meno precisi a seconda del grado di accuratezza desiderato.

Per quanto riguarda i video, la descrizione delle entità "sequenze video" dipende dal contesto del filmato in esame e dall'ambito cognitivo in cui viene utilizzata. Per questi motivi, la fase di indicizzazione dei documenti deve sfruttare il contenuto della sequenza video in base alle caratteristiche tipiche del dominio applicativo. Per esempio, può essere interessante effettuare ricerche per soggetto (singolo attore, particolare scenografia), tipo di inquadratura (primo piano, quadro d'insieme), tipo di azione (entrata in scena, uscita dalla scena, coro d'insieme).

Per facilitare la navigazione e il reperimento di sequenze video digitali, è utile descrivere i filmati in termini di dati strutturati e riconoscere le interruzioni logiche al loro interno. Tale tecnica è detta *shot boundary detection* e il suo compito è quello di segmentare lunghe sequenze, in modo automatico, sfruttando una analisi basata sul contenuto. Il rilevamento e la classificazione di interruzioni di scene è un primo passo nel processo di annotazione automatica di sequenze video digitali. Una volta eseguita la segmentazione, il video originale può essere pensato come suddiviso in porzioni internamente uniformi e ininterrotte, sulle quali è possibile effettuare ricerche più specifiche.

7. CONCLUSIONI E FUTURI SVILUPPI

Nella prefazione del libro *Pattern Recognition* del 1974, Vapnik e Chervonenkis scrissero (l'originale è in russo): "...Per costruire la teoria della *pattern recognition*, prima di tutto è necessario formalizzare completamente il problema del riconoscimento automatico di oggetti (...) e questo risultato sembra essere difficile da raggiungere."

Come abbiamo visto nei paragrafi precedenti le tecniche di *pattern recognition* sono molto dipendenti dal dominio, ed anche le reti neurali che sembrano un approccio più generale in realtà dipendono dall'insieme di apprendimento (training set) e dai parametri degli algoritmi di back-propagation usati nell'addestramento (per esempio: il learning rate).

Essenzialmente, i differenti punti di vista sulla formulazione del problema della *pattern recognition* sono rappresentati dalla risposta

ad una domanda: esistono principi generali adeguati per descrivere classi di pattern di diversa natura, oppure lo sviluppo di un linguaggio di descrizione dei pattern è un problema che va affrontato dai singoli esperti per ogni dominio in ogni singolo caso?

Se la prima parte della domanda fosse vera, allora la scoperta di questi principi dovrebbe essere la principale direzione della ricerca nella *pattern recognition*. [7].

Altrimenti la *pattern recognition* si ridurrebbe a minimizzare l'errore nell'assegnare il pattern alla classe sbagliata, o *risk minimization*; questo implicherebbe la definizione di regioni di decisione specifiche per ogni classe di pattern e diventerebbe un problema di statistica applicata.

Finora le risposte alla domanda fondamentale non sono state trovate, e questo è il motivo per cui formulare un modello di *pattern recognition* è ancora un problema arduo. La maggior parte dei ricercatori ha comunque adottato il secondo punto di vista e la teoria della *pattern recognition* è attualmente legata strettamente alla teoria della *risk minimization* con una classe di regioni di decisioni dipendenti dal dominio.

Alla fine degli anni cinquanta, i primi ricercatori sulla *pattern recognition* cercavano di formulare nuovi modelli e non di ridurre i problemi a schemi matematici o statistici noti. Infatti in quest'ottica quello che attualmente accade, e cioè di ridurre i problemi di *pattern recognition* a quelli di *risk minimization*, ha sollevato non poco disappunto nelle attese dei ricercatori. A quell'epoca però molti dividevano la sensazione che si potesse trovare qualcosa di completamente nuovo per modellare i problemi di *pattern recognition*.

Attualmente, dopo più di quarant'anni, è chiaramente naturale chiedersi se l'importanza della modellazione formale nella rappresentazione dei pattern è stata pienamente compresa. Evidentemente la risposta è no, poiché l'atteggiamento di riutilizzare schemi matematici noti continua a prevalere.

Naturalmente va detto che non c'è stato un grandissimo sforzo in questo senso per arrivare ad un formalismo unificante. Negli anni '70 ed '80, per esempio, i nuovi approcci sintattici e strutturali alla *pattern recognition* diedero un ruolo centrale alla rappresenta-

zione non numerica dei pattern, e ci si aspettava che il futuro del campo fosse connesso con l'integrazione degli approcci strutturali con quelli classici, basati sui *vector spaces* (spazi di vettori di *feature*). Ma questo non è ancora avvenuto, o meglio non sembra sufficientemente unificante.

Siamo in realtà ad una svolta cruciale nel campo della *pattern recognition* in quest'ultimo periodo; infatti, c'è la necessità di produrre inquadramenti formali che generalizzino l'approccio in modo da evidenziare la natura intrinsecamente "intelligente" dell'attività di *pattern recognition*. Basandoci su questo vanno definiti i concetti chiave su cui poggiare queste infrastrutture formali o framework. Come delineato nel workshop satellite dell'International Conference on Pattern recognition del 2004, dal titolo *Pattern representation and the future of pattern recognition*, va compreso che l'investimento di ingenti risorse nello sviluppo di nuove tecniche basate sul vecchio approccio (riduzione a statistica applicata) è irrazionale comparato a quello necessario per costruire un impianto formale adeguato. Infatti, è da sottolineare il fatto che la Scienza è orientata al futuro e quindi alla scoperta e costruzione degli scenari generali o "big picture" e non alla risoluzione indipendente di problemi, per quanto essi siano complicati.

Gli sviluppi futuri, quindi, si stanno orientando verso la *inductive class representation* che, descritta in poche parole, è un approccio sia generativo che induttivo alla pattern recognition. Il fatto di essere un approccio generativo implica che ha la capacità di "generare" oggetti dalle classi basandosi sulla rappresentazione formale delle stesse. L'induzione nell'approccio è intesa come la possibilità che la rappresentazione delle classi possa essere dedotta da un piccolo insieme di esempi (*training set*) e che tale rappresentazione sia stabile rispetto ai vari tipi di "rumore" ed errori presenti a seconda della rappresentazione scelta.

I ricercatori propongono attualmente un approccio chiamato ETS, *Evolving Transformation System*. L'ETS, differentemente dai classici modelli computazionali (per esempio, linguaggi e grammatiche formali) evidenzia la natura evolutiva delle classi di oggetti e

quindi della loro rappresentazione. Queste strutture matematiche permettono di rappresentare dinamicamente, durante il processo di apprendimento, la struttura composta degli oggetti all'interno di un ambiente induttivo o evolutivo. In altre parole, il modello è basato su un nuovo tipo di struttura per la rappresentazione (induttiva) degli oggetti, chiamata *struct* che è definita come un elemento di uno spazio generalizzato di misure, su cui la misura di similarità (tra oggetti) è rappresentata da un insieme (ottimo) finito di trasformazioni strutturali pesate (per esempio, come le operazioni di inserimento sensibili al contesto) costruite durante il processo di apprendimento.

Si può infine notare che questa scelta sta aprendo realmente nuove strade per la ricerca dei prossimi anni, poiché il concetto di rappresentazione evolutiva di classe non può essere integralmente introdotto nei tipici formalismi per via delle loro limitazioni. Per esempio nel formalismo numerico, la rappresentazione di una classe non è generativo, mentre nei linguaggi formali (comprese le grammatiche su grafi), il concetto di classe non è induttivo; questo discende dal fatto che quest'ultimo è un formalismo computazionale (logico) e non basato sulla rappresentazione.

Per concludere segnaliamo la presenza, sia in Italia che nel mondo, di molti centri di ricerca, sia accademici che industriali, fortemente orientati alle applicazioni di riconoscimento automatico. Tra i principali attori in ambito accademico internazionale possiamo citare: il Machine Intelligence Laboratory dell'Università di Cambridge (UK), il Computer Vision Lab dell'Università Linköping (Svezia), il centro di ricerca LORIA (Francia), l'Intelligent Systems Lab di Amsterdam (Olanda), il MIT Media Lab (USA), il Rensselaer Polytechnic Institute (USA), il Computer Vision Lab dell'Università del Maryland (USA), il gruppo di Pattern Recognition and Image Processing dell'Università del Michigan, MSU (USA), il Pattern Recognition Lab della Lehigh University (USA).

A livello industriale i principali centri di ricerca sono: il PARC (Palo Alto Research Center, USA), il Vision Technology Group della Microsoft (Cambridge, UK), l'Exploratory Computer Vision Group di IBM (USA).

Nel panorama nazionale sono presenti molti gruppi accademici di fama internazionale distribuiti su tutto il territorio dal nord al sud, isole comprese. Esistono, inoltre, numerose riviste internazionali dedicate a queste tematiche, tra le principali menzioniamo: *The Journal of the Pattern Recognition Society* edito dalla Elsevier, e *Transaction on Pattern Analysis and Machine Intelligence* edito dalla IEEE. Le conferenze di riferimento sono l'IC-PR, ovvero *International Conference on Pattern Recognition* che si svolge ogni due anni a livello internazionale; mentre a livello nazionale (si svolge sempre in Italia ma con respiro internazionale) è ormai ben conosciuta l'ICIAP, *International Conference on Image Analysis and Processing*, che include molte tematiche di Pattern Recognition.

Bibliografia.

- [1] Biagiola L.: *Riconoscimento automatico di soggetti in movimento*. Tesi di Laurea, Dipartimento di Informatica, Università "La Sapienza" di Roma, Italia, A.A. 2002-2003.
- [2] Devroye L., Györfi L., Lugosi G.: *A Probabilistic theory of Pattern Recognition*. Berlin: Springer-Verlag, 1996.
- [3] Duda R.O., Hart P., Stork D.G.: *Pattern Classification and Scene Analysis*. New York: John Wiley & Sons, 2000.
- [4] Duda R.O., Hart P.: *Pattern Classification and Scene Analysis*. New York: John Wiley & Sons, 1973.
- [5] Fu K.S.: A step toward unification of Syntactic and Statistical Pattern Recognition. *IEEE Transaction on Pattern Recognition and Machine Intelligence*, Vol.5, n. 2, 1983, p. 200-205.
- [6] Fu K.S.: *Syntactic Pattern Recognition and Applications*. Englewood Cliffs, N.J., Prentice-Hall, 1982.
- [7] Lev Goldfarb, David Gay, Oleg Golubitsky, Dmitry Korkin: *What is a structural representation? Second variation*. Faculty of Computer Science, U.N.B., Technical Report TR04-165, March 2004.
- [8] Grenander U.: *General Pattern Theory*. Oxford University Press, 1993.
- [9] Jain A., Duijn R., Mao J.: Statistical Pattern Recognition: A Review. *IEEE Transaction on Pattern Recognition and Machine Intelligence*, Vol. 22, n. 1, 2000, p. 4-37.
- [10] Jain A., Mao J., Mohiuddin M.K.: *Artificial Neural Networks: A Tutorial*. Computer, Mar. 1996, p. 31-44.
- [11] Kohonen T.: *Self Organizing Maps*. Springer Series in Information Sciences, Vol. 30, Berlin, 1995.
- [12] El-Kwae E.A., Kabuka M.R.: Efficient content-based indexing of large image databases. *ACM Trans. Inf. Syst.*, Vol. 18, n. 2, Apr. 2000, p. 171-210.
- [13] Ngu A.H., Sheng Q., Huynh D., Lei R.: Combining multi-visual features for efficient indexing in a large image database. *The VLDB Journal*, Vol. 9, n. 4, Apr. 2001, p. 279-293.
- [14] Pavlidis T.: *Structural Pattern Recognition*. New York, Springer-Verlag, 1977.
- [15] Perlovsky L.I.: Conundrum of Combinatorial Complexity. *IEEE Transaction on Pattern Recognition and Machine Intelligence*, Vol. 20, n. 6, 1998, p. 666-670.
- [16] Picard R.: *Affective Computing*. MIT Press, 1997.
- [17] Vapnik V.N.: *Statistical Learning Theory*. New York: John Wiley & Sons, 1998.
- [18] Watanabe S.: *Pattern Recognition: Human and Mechanical*. New York: Wiley, 1985.
- [19] Wikipedia - the free encyclopedia, <http://www.wikipedia.org>.

ALESSIO MALIZIA si è laureato in Scienze dell'Informazione presso l'Università "La Sapienza" di Roma nel 1997, dove attualmente svolge la sua attività di ricerca. La sua ricerca comprende il riconoscimento automatico delle forme, le biblioteche digitali, e la visione artificiale. Attualmente lavora su Interazione uomo-macchina, modelli collaborativi, ed interazione multimodale. Nella sua passata esperienza ha lavorato per IBM e Silicon Graphics nell'ambito delle applicazioni multimediali. È stato visiting researcher presso lo XEROX PARC (Palo Alto Research Center) dove ha lavorato nel gruppo di Human-Document Interaction.
E-mail: malizia@di.uniroma1.it