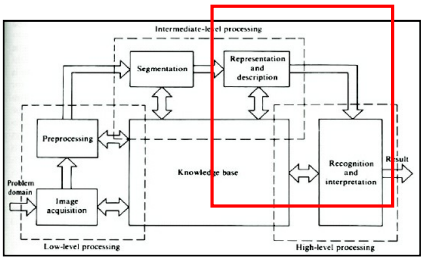
Introduzione alla classificazione

Raimondo Schettini

Universita' di Milano Bicocca
Raimondo.Schettini@unimib.it
www.ivl.disco.unimib.it/Schettini/

1

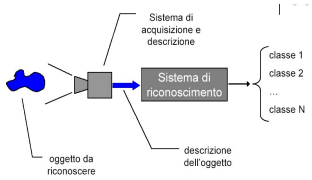
Elaborazione delle immagini



2

Classificazione

Scopo: definire un sistema per riconoscere automaticamente un oggetto o evento.



data la descrizione di un oggetto che può appartenere ad una tra N classi possibili, compito del sistema è attribuire l'oggetto ad una classe, utilizzando una base di conoscenza precedentemente costruita.

Pattern Classification (2nd ed) by R. O. Duda, P. E. Hart and D. G. Stork, John Wiley & Sons, 2000

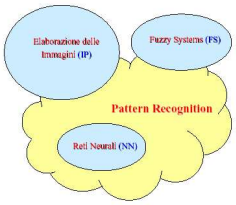
3

Classificazione

La classificazione ha il compito di associare ad ogni pattern una classe compresa in un insieme di classi predefinite. Un termine a volte usato al posto di Classificazione è Riconoscimento.

Molto usata in:

- Applicazioni industriali
- Analisi di immagini biomediche
- Analisi di immagini tele-rilevate
- Riconoscimento di caratteri ottici (OCR)
- Data mining
- Identificazione delle impronta digitale
- Identificazione di una sequenza di DNA
- Riconoscimento di segnali vocali



4

Classificazione

Dominio problema	Applicazione	Input: Pattern	Output: Classe
Analisi di documenti	OCR: conversione di immagini in testo	Immagini di documenti	Caratteri alfanum., parole
Automazione industriale	Ispezione di circuiti stampati	Immagine del circuito	Difettoso / Non difettoso
Bioinformatica	Analisi delle sequenze	Sequenza DNA/Proteine	Tipo conosciuto di gene
Classificazione documenti	Ricerca su Internet	Documento di testo	Categoria semantica
Data mining	Ricerca di pattern "significativi"	Punti multi-dimensionali	Cluster compatti e ben separati
Database Immagini	Ricerca su database immagini	Collezioni di immagini	Specifici soggetti o temi
Economia	Predizione mercato azionario	Sequenze storiche	Acquista / Vendi

5

Classificazione

Dominio problema	Applicazione	Input: Pattern	Output: Classe
Medicina	Analisi immagini radiografiche	Immagine alta risoluzione	Sano / Malato
Militare	Abbattimento automatico missili	Immagini "live"	Direzione di calibrazione
Riconoscimento del parlato	Risponditori telefonici autom.	Segnale audio del parlato	Parole dette
Sistemi Biometrici	Riconoscimento di persone	Volto, Iride, Impronta digitale	Utente autorizzato
Sorveglianza	Sistema anti-intrusione	Immagine "live" del locale	Normale / Allarme
Telerilevamento	Stimare densità di colture	Immagini multispettrali	Tipi di coltivazioni
Visione robotica	Guida automatica di un veicolo	Immagini "live"	Direzione sterzo

6

Classificazione

- Assunzione: Oggetti simili producono segnali simili

- Una CLASSE è un insieme di oggetti simili.
- I PATTERN sono l'unione dei segnali che provengono da oggetti simili.
- PATTERN CLASSIFICATION è il processo di identificazione del segnale come originato da una particolare classe di oggetti
- CLASSIFICATORE: è un sistema che dato un oggetto lo classifica rispetto a classi predefinite.
- FEATURE: insieme delle caratteristiche che descrivono un oggetto

7

Progetto di applicazioni di classificazione

TEORIA

- Specificare uno spazio di interpretazione (lo spazio delle classi, cosa vogliamo classificare ?)

- Dare una descrizione appropriata degli oggetti, fisici o concettuali, in termini di caratteristiche (feature), appartenenti a uno spazio di rappresentazione.

- Fornire un mapping fra lo spazio di rappresentazione e quello di interpretazione.

- Determinare un operatore (algoritmo) che esegua questo mapping, ovvero che esegua la classificazione.

8

Progetto di applicazioni di classificazione

PRATICA

- Comprendere il problema
- Raccogliere una casistica significativa ed analizzarne le caratteristiche.
- Studiare la letteratura per vedere come sono stati affrontati casi simili.

  – Quali feature e quale metodo usare?

- Sperimentare.
- Valutare oggettivamente i risultati
- Documentare gli esperimenti fatti
- Raffinare / Iterare il processo

9

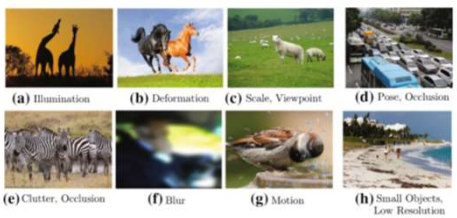
Comprendere lo scopo della Classificazione

I concetto di classe dipende dalla semantica dell'applicazione stesso insieme:
- 2 classi per distinguere le lettere dell'alfabeto dai numeri
- 21 classi per il riconoscimento delle lettere

Le classi possono essere definite dal progettista, si parla quindi di classificazione supervisionata, o "apprese" automaticamente dal sistema, classificazione non supervisionata. Se un pattern viene associato a una sola classe si parla di ***classificazione esclusiva***; se invece il pattern può appartenere, con un certo grado di probabilità, a più classi si parla di ***classificazione continua o fuzzy***.

10

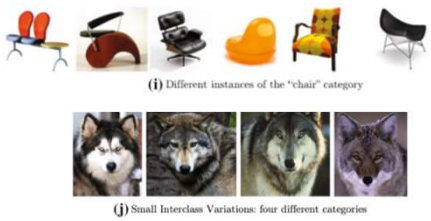
Comprendere i dati



Quanto sono variabili le condizioni di acquisizione ?

11

Comprendere i dati



Quanto sono diversi gli oggetti delle stessa classe, variabilità intra-classe, e quando sono simili le classi variabilità inter-classe ?

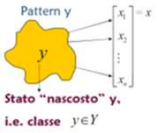
12

Feature

• Le Feature sono espressioni numeriche delle proprietà di un segnale. L'insieme di feature usate da un sistema PR è chiamato feature vector.

•Il numero di feature impiegate è la dimensione del vettore delle feature.

•Vettori di feature n-dimensionali possono essere rappresentati come punti nello spazio n-dimensionale delle feature. Spazio delle feature.



13

Feature

Bisogna scegliere quali caratteristiche (feature) dell'oggetto bisogna da misurare.

Criteri:

– discriminazione: valori differenti per differenti classi;

– affidabilità: valori simili per classi uguali

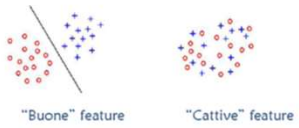
– indipendenza: feature scorrelate tra loro (se correlate meglio fonderle insieme che usarle separate)

– dimensione: poche feature limitano la complessità del classificatore

14

Selezione delle feature (*feature selection*)

- Usare feature che ben discriminino tra classi permette di
  1. Ridurre il numero di esempi del training set.
  2. Aumenta la qualità della funzione di riconoscimento



15

Selezione delle feature (*feature selection*)

• Selezione delle feature: Tra tutte le feature calcolabili, scegliere l'insieme minimo discriminante ed affidabile che permette di ottenere le prestazioni desiderate.

-> abbiamo bisogno di un metodo per valutare le prestazioni! -> i.e. errore

• Selezione delle feature: Approccio combinatorio

– M feature (totali), N feature (ridotte), N < M
– testare tutte le possibili combinazioni di N feature per addestrare il classificatore
– calcolare l'errore per ogni sottoinsieme
– scegliere quello che porta all'errore più piccolo.

16

Selezione delle feature (*feature selection*)

– date 2 feature x e y; training set di oggetti da M classi, N_j numero di oggetti classe j; x_{ij}, y_{ij} valore feature dell'i-esimo oggetto $\in$ classe j.
– Calcolo delle medie (stimati sulla base del training set):

$$\hat{\mu}_{x_j} = \frac{1}{N_j}\sum_{i=1}^{N_j} x_{ij} \qquad \hat{\mu}_{y_j} = \frac{1}{N_j}\sum_{i=1}^{N_j} y_{ij}$$

– Calcolo varianze:

$$\hat{\sigma}_{x_j} = \frac{1}{N_j}\sum_{i=1}^{N_j}\left(x_{ij} - \hat{\mu}_{x_j}\right)^2 \qquad \hat{\sigma}_{y_j} = \frac{1}{N_j}\sum_{i=1}^{N_j}\left(y_{ij} - \hat{\mu}_{y_j}\right)^2$$

– Calcolo correlazione di x e y nella classe j:

$$\hat{\sigma}_{xy_j} = \frac{\frac{1}{N_j}\sum_{i=1}^{N_j}\left(x_{ij} - \hat{\mu}_{x_j}\right)\left(y_{ij} - \hat{\mu}_{y_j}\right)}{\hat{\sigma}_{x_j}\hat{\sigma}_{y_j}}$$

17

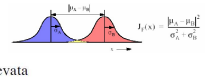
Selezione delle feature (*feature selection*)

dove $-1 \le \hat{\sigma}_{xy_j} \le 1$

$\hat{\sigma}_{xy_j} \equiv 0 \Rightarrow$ feature scorrelate
$\hat{\sigma}_{xy_j} \equiv 1 \Rightarrow$ feature fortemente correlate (meglio combinarle o scartarne una)
$\hat{\sigma}_{xy_j} \equiv -1 \Rightarrow$ feature correlate ma inversamente proporzionali (meglio combinarle o scartarne una)

– Distanza di separazione tra le classi j e k per la feature x:

$$\hat{D}_{x_{jk}} = \frac{\left|\hat{\mu}_{x_j} - \hat{\mu}_{x_k}\right|}{\sqrt{\hat{\sigma}^2_{x_j} + \hat{\sigma}^2_{x_k}}}$$



e si scegli la feature che produce la distanza più elevata

18

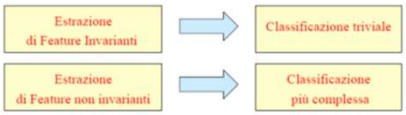
Selezione delle feature (*feature selection*)

- L'onere computazionale della classificazione cambia al variare del numero di feature a disposizione con un andamento che dipende del tipo di classificatore.
- La riduzione delle feature è motivata da due aspetti principali:
  - # Minimizzare il costo realizzativo del sistema di riconoscimento (riducendo il numero di misure da effettuare e, di conseguenza, l'onere computazionale comportato dall' elaborazione delle misure).
  - # Ridurre i problemi di stima delle statistiche delle classi informative dovuti al numero limitato di campioni di training disponibili.
- Si cerca un buon compromesso tra numero di feature ed accuratezza di classificazione. Esistono due approcci principali alla riduzione delle feature:
  - # Ordinamento e selezione delle feature in base alla loro capacità di discriminare le diverse classi informative
  - # Trasformazione dello spazio delle feature

19

Caratteristiche delle feature vs classificatore

Con il termine feature invarianti si denotano caratteristiche estratte dai pattern che siano costanti (il più possibile) rispetto alle possibili variazioni intra-classe

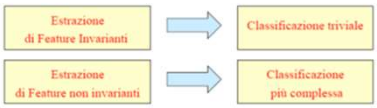


Uno dei maggiori dilemmi in applicazioni di PR è se l'invarianza dei pattern debba essere gestita a livello di feature (attraverso la scelta e l'estrazione di feature invarianti) o a livello di classificazione (attraverso la scelta e il progetto di metodi in grado di sopportare feature non invarianti).

20

Caratteristiche delle feature vs classificatore

- Generalmente se l'invarianza è gestita a livello di feature, la classificazione è molto più veloce; questo può far propendere per tale soluzione nel caso ad esempio di riconoscimento rispetto a un elevato numero di classi (es: *identificazione di un individuo su un database di 1 milione di individui*).
- Spesso la gestione dell'invarianza a livello di feature è più difficile, e i metodi che utilizzano classificatori robusti (e feature non invarianti) forniscono maggiore accuratezza e affidabilità.



21

Normalizzazione delle feature

Secondo Wikipedia, *"Il feature scaling è un metodo utilizzato per standardizzare l'intervallo di variabili indipendenti o caratteristiche dei dati. Nell'elaborazione dei dati, è anche nota come normalizzazione dei dati e viene generalmente eseguita durante la pre-elaborazione dei dati."*

Il valore Z-standard viene calcolato come segue, ove u è la **media** dei campioni di addestramento, e σ è la **deviazione standard** dei campioni di addestramento e x_i è il valore che si vuole standardizzare.

$$z = \frac{x_i - u}{\sigma}$$

Un metodo alternativo alla standardizzazione è la **normalizzazione** (o **Min-Max Scaling**)

$$z = \frac{x - \min(x)}{\max(x) - \min(x)}$$

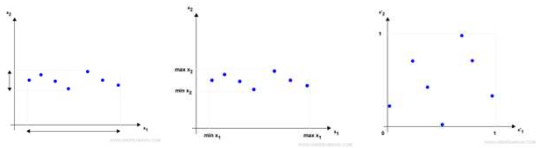
In alcune situazioni, si può preferire mappare i dati su un intervallo [-1,1]

$$z = \frac{x - \text{media}(x)}{\max(x) - \min(x)}$$

22

Normalizzazione delle feature

Uno dei più semplici è il ridimensionamento min-max.



$$x'_i = \frac{x_i - min(x)}{max(x) - min(x)}$$

23

Principali approcci della Pattern Recongnition

1.Template matching
Idea : costruire uno o più pattern modello (template) e "cercarlo/i" all'interno dell'immagine misurando il grado di "matching" nelle diverse possibili posizioni.

2. Approccio Statistico
Ogni pattern è rappresentato da un punto nello spazio multi-dimensionale.
Prevede una fase di estrazione delle caratteristiche (feature) che mappa il pattern nel punto e una fase di classificazione che associa il punto a una classe.
I classificatori utilizzati sono fondati su solide basi statistiche.

24

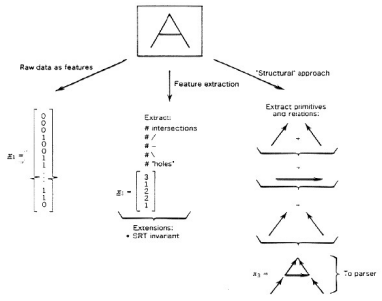
I 4 principali approcci del PR

3. Approccio Strutturale (sintattico):
I pattern sono codificati in termini di componenti primitive e di relazioni che intercorrono tra esse. Il confronto avviene confrontando primitive e relazioni.

4. Reti Neurali:
Sono costituite da grafi orientati i cui nodi (neuroni) processano le informazioni trasmesse da altri neuroni ad essi collegati. Consentono di "codificare" complessi mapping non-lineari, che vengono solitamente "appresi" da esempi.

25

Classificazione: esempio 1



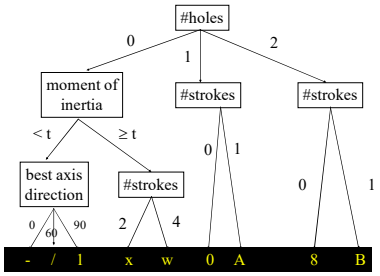
26

Classificazione: esempio 1

(class) character	area	height	width	number #holes	number #strokes	(cx,cy) center	best axis	least inertia
'A'	medium	high	3/4	1	3	1/2,2/3	90	medium
'B'	medium	high	3/4	2	1	1/3,1/2	90	large
'8'	medium	high	2/3	2	0	1/2,1/2	90	medium
'0'	medium	high	2/3	1	0	1/2,1/2	90	large
'1'	low	high	1/4	0	1	1/2,1/2	90	low
'W'	high	high	1	0	4	1/2,2/3	90	large
'X'	high	high	3/4	0	2	1/2,1/2	?	large
'*'	medium	low	1/2	0	0	1/2,1/2	?	large
'-'	low	low	2/3	0	1	1/2,1/2	0	low
'/'	low	high	2/3	0	1	1/2,1/2	60	low

27

Classificazione: esempio 1



28

Classificazione: esempio 2

Separare i pesci (a secondo la specie) in una catena di convogliamento.
Due classi: Salmone (Salmon) e Spigola (Sea bass).

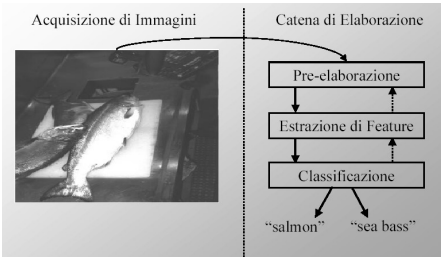
Si può utilizzare come sensore una telecamera e progettare una catena di elaborazione per il riconoscimento del contenuto delle immagini acquisite.
Delle immagini campione servono per estrarre e capire quali sono le potenziali feature da considerare per la nostra applicazione:

- Lunghezza
- Larghezza
- Chiarezza
- Posizione della bocca

•L'insieme delle immagini campione è detto insieme di addestramento o training set

29

Classificazione: esempio 2



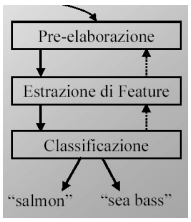
30

Classificazione: esempio 2

Pre-processing/segmentazione per isolare i pesci tra loro e dallo sfondo.

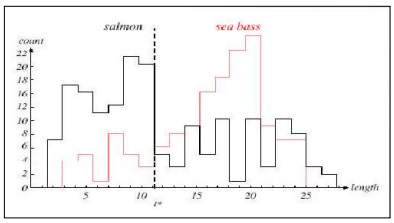
Estrazione di feature:
estrarre dalle immagine di ciascun pesce le informazioni (feature) più rilevanti per discriminare (distinguere) al meglio possibile i pesci tra loro; ad esempio, si può scegliere la lunghezza come possibile feature per la discriminazione tra i pesci.

Classificazione:
sulla base delle feature selezionate, si decide quale è la classe dell'oggetto osservato (salmon oppure sea bass.)



31

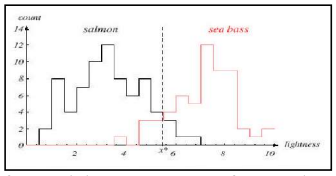
Classificazione: esempio 2



Come si puo' vedere dall'istogramma, usare la lunghezza per discriminare i due tipi di pesce (classi) darebbe risultati non soddisfacenti. Si dice chelLa feature scelta non è discriminante. C'è una certa differenza, in media, ma non tale da separare nettamente le due classi.

32

Classificazione: esempio 2



La chiarezza è una feature più discriminate. La nuova feature scelta permette una distinzione migliore tra le due classi. La soglia x* è scelta in modo da minimizzare l'errore totale. Con feature mono-dimensionali il processo di classificazione si riduce ad una sogliatura ottima, nel senso che minimizzi un costo predefinito (ad esempio la perdita economica). Nel esempio la soglia è scelta ritenendo uguale il costo dei due tipi di errore.
Ipotesi spesso non valida.

33

Errori di classificazione

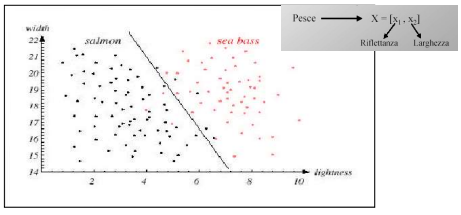
A rigore, il costo non è associato alla classificazione erronea, ma alla decisione che viene presa in base a quella classificazione.

Classificare una *Tigre* come *Giaguaro* o come *Gatto* è sempre un errore, ma può costare diversamente.

- Dato un test rapido COVID è più grave dire che un soggetto è positivo anche se non lo è o il contrario ?

34

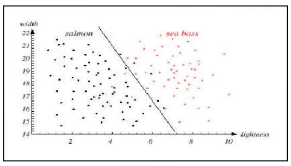
Classificazione: esempio 2



Potremmo in realta' possono usare entrambe le feature (lunghezza e chiarezza), e si potrebbero aggiungere altre feature scorrelate.
Si deve fare attenzione a non ridurre le prestazioni del classificatore introducendo feature rumorose.

35

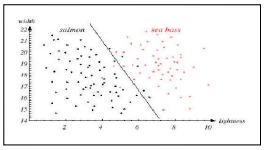
Classificazione: esempio 2



- Il problema ora è dividere lo spazio delle features in regioni, ognuna delle quali sia ascrivibile ad una delle classi note.
- Si identificano così delle *regioni di decisione* (*decision regions*), separate da una frontiera (*decision boundary*).
- In questo modo è possibile decidere a quale classe assegnare il campione sulla base della posizione del punto nel feature space.

36

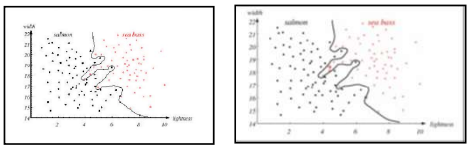
Classificazione: esempio 2



- La scelta più immediata è quella di una frontiera semplice, lineare. Gli errori complessivi sono minori rispetto al caso di una sola feature, ma sono comunque presenti.
- Sarebbe possibile eliminare del tutto gli errori con una frontiera meno semplice.
  - La frontiera di decisione è generata dal sistema di classificazione; quindi una frontiera meno semplice implica un classificatore più complesso. Ma siamo sicuri che sarebbe veramente migliore?

37

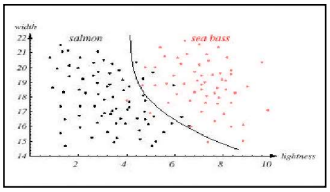
Classificazione: esempio 2



Nel caso ideale, la migliore frontiera di decisione dovrebbe essere quella che fornisce le prestazioni ottime (nessun errore di classificazione) come mostrato nella figura a sinistra.
Come verrà classificato un nuovo campione nella regione (?): verrà classificato sea bass ma più probabilmente è un salmone. Abbiamo sovra-modellato (over-fitting) il training set.
Nella pratica si cerca di evitare l'over-fitting dei dati di apprendimento.

38

Classificazione: esempio 2



Una frontiera di decisione più complessa della frontiera lineare. Sebbene gli errori sul training set siano ancora presenti, il classificatore sembra garantire una buona capacità di generalizzazione.

39

Generalizzazione del classificatore

- E' improbabile che un classificatore estremamente complesso garantisca buone capacità di generalizzazione in quanto costruito strettamente sulle caratteristiche dei campioni del particolare training set (e del particolare rumore che si portano dietro).
- Un classificatore efficace dovrebbe invece essere costruito su caratteristiche e strategie generali che siano valide anche per campioni non appartenenti al training set.
- Si impone quindi di stabilire un compromesso tra:
  - prestazioni del classificatore sul training set
  - capacità di generalizzazione del classificatore su dati non visti. Test set
- Di conseguenza, è preferibile tollerare qualche errore sul training set se questo porta ad una migliore generalizzazione del classificatore.

40

Training set e test set

Insieme di Campioni di Training
Insieme di campioni per i quali la classe d'appartenenza è nota. Questi campioni sono usati per trovare la frontiera di decisione ottima, cioè per progettare (addestrare) il classificatore;

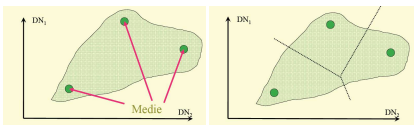
Problema di Generalizzazione
Lo scopo del classificatore è essere capace di riconoscere ogni campione incognito (classe d'appartenenza non nota) con il margine di errore più piccolo possibile.
Quindi, bisogna stare attenti al fatto che il classificatore non sia troppo adattato ai campioni di training (problema dell'overfitting).

Insieme di Campioni di test
Insieme di campioni per i quali la classe d'appartenenza è nota è che non sono stati nel training . Tale insieme e' usato per valutare le performance del classificatore

41

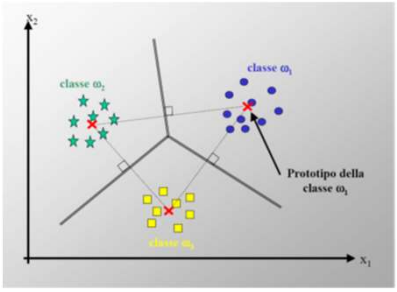
Classificatore a Minima Distanza

Un classificatore banale che abbiamo già usato per la segmentazione ed il clustering è il classificatore a minima distanza.
In un classificatore a minima distanza ciascuna classe ha una media associata (vettore delle features $\mathbf{m}_j$) . Assegno a un vettore $\mathbf{x}$ la classe j tale che $D_j = ||\mathbf{x}-\mathbf{m}_j||$ sia minimo (distanza euclidea).
Funziona solo se tutte le feature sono discriminanti e nello stesso modo! -> necessità di normalizzare. Si devono stimare le medie (training) e definire una funzione distanza.
La partizione dello spazio dei parametri che rappresenta le classi è un diagramma di Voronoi.



42

Classificatore a Minima Distanza



43

Classificatore a Minima Distanza

- Algoritmo

1) Calcolare il baricentro $\mathbf{b}_i$ (i = 1, 2, …, C) di ciascuna delle C classi informative;

2) Per classificare un campione sconosciuto $X=[x_1,x_2,\ldots,x_N]$ dello spazio delle feature di dimensione N, scegliere la classe ω^* che soddisfa la seguente condizione:

$$\omega^* = \underset{i=1,2,\ldots,C}{\operatorname{argmin}} \{d(X, \mathbf{b}_i)\}$$

dove d(.) rappresenta un'opportuna funzione di distanza.

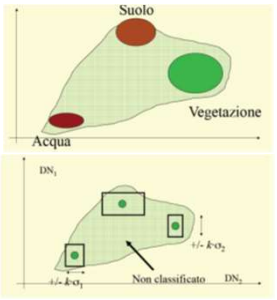
44

Classificatore a Minima Distanza

- In questo metodo, si suppone:
  - che i campioni abbiano poco variabilità attorno ad un pattern tipico e rappresentativo della classe;
  - oppure che tutte le classi abbiano lo stesso andamento statistico (stessa matrice di covarianza);
- Ciascuna classe verrà modellata nello spazio delle feature tramite il suo vettore medio (statistica di 1° ordine) che giocherà il ruolo del prototipo della classe. Non tiene conto della variabilità statistica delle classi
- La classificazione di una data osservazione (campione sconosciuto) verrà fatta sulla base della minima distanza tra l'osservazione ed i prototipi delle classi. Semplice, onere computazionale basso
- Un classificatore basato sulla minima distanza è caratterizzato da frontiere di decisione lineari.

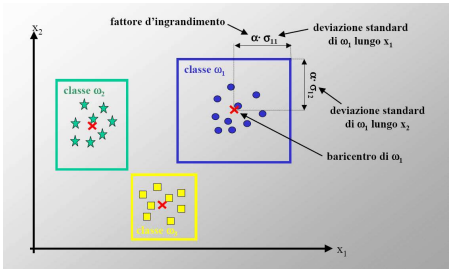
45

Classificatore a parallelepipedo



46

Classificatore a parallelepipedo



47

Classificatore a parallelepipedo

1) Calcolare il baricentro $\mathbf{b}_i$ (i = 1,…, C) di ciascuna delle C classi informative;

2) Calcolare la deviazione standard σ_{ij} di ciascuna delle classi ω_i (i = 1,…, C) lungo ciascuna delle feature x_j (j = 1,…, N);

3) Per classificare un campione sconosciuto $X=[x_1,x_2,\ldots,x_N]$ dello spazio delle feature di dimensione N, scegliere la classe ω_k che soddisfa la seguente condizione:

$$b_{kj} - \alpha.\sigma_{kj} \le x_j \le b_{kj} + \alpha.\sigma_{kj}, \quad \forall\, j = 1,\ldots,N$$

Alpha è il fattore di ingrandimento. Se si verifica il caso in cui l'osservazione non appartiene a nessuna regione di decisione (rettangolo), allora essa non verrà etichettata. Il fattore α viene spesso scelto in modo da minimizzare il numero di osservazioni rimaste non classificate.
Al limite, si può utilizzare il massimo valore del fattore d'ingrandimento al di sopra del quale almeno due regioni iniziano a sovrapporsi.

48

Classificatore a parallelepipedo

- Al contrario del metodo della minima distanza, il metodo del parallelepipedo ("*Box Classifier*") tiene conto, anche se in modo molto primitivo, della statistica di secondo ordine delle classi.
- Questo metodo modella ciascuna classe nello spazio delle feature con una densità di probabilità uniforme contenuta in un rettangolo multidimensionale:
  - centrato nel suo baricentro;
  - di dimensioni espresse in funzione delle deviazioni standard della classe lungo ciascuna direzione delle feature;
- In pratica, i rettangoli multidimensionali possono essere visti come le regioni di decisione associate alle classi.
- Una data osservazione verrà assegnata ad una classe se appartiene alla sua regione di decisione. Semplice, onere computazionale basso.

49

Classificatore Nearest Neighbour (NN)

•Data una metrica d(.) nello spazio multidimensionale (es. distanza euclidea) il classificatore nearest-neighbor (letteralmente "il più vicino tra i vicini"), assegna un pattern *x* alla stessa classe dell'elemento *x'* ad esso più vicino nel training set.

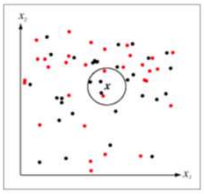
•La regola NN produce una tassellazione di Voronoi: Ogni elemento x_i del TS determina un tassello, all'intero del quale i pattern saranno assegnati alla stessa classe di x_i.

•Basta che un solo elemento del training set non sia molto "affidabile" (outlier) affinché tutti i pattern nelle sue vicinanze siano etichettati non correttamente.

•Un modo generalmente più robusto, che può essere visto come estensione della regola NN (in questo caso detta 1-NN) è il cosiddetto classificatore *k*-nearest-neighbor (*k*-NN).

50

Classificatore Nearest Neighbour (NN)



La regola k-NN determina i k elementi più vicini al pattern x da classificare; ogni pattern tra i k vicini "vota" per la classe cui esso stesso appartiene; il pattern x viene assegnato alla classe che ha ottenuto il maggior numero di voti.

Nell'esempio, Il classificatore 5-NN assegna **x** alla classe "nera", in quanto quest'ultima ha ricevuto 3 voti su 5.

k dispari per cercare di evitare "pareggi".

Per questo metodo, è quindi necessario memorizzare tutti I campioni di training.
Per la scelta del valore del parametro k, non esiste un metodo teorico per stimarlo. Questo valore dipende molto da come sono distribuite e sovrapposte le classi nello spazio delle feature.

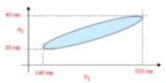
51

Classificatore Nearest Neighbour (NN)- caso studio

- Il comportamento di un classificatore è strettamente legato alla metrica (funzione distanza) adottata.
- La distanza euclidea, che rappresenta il caso L2 nella definizione di metriche di Minkowski, è sicuramente la metrica più spesso utilizzata.
- Nella pratica, prima di adottare semplicemente la distanza euclidea è bene valutare lo spazio di variazione delle componenti (o feature) e la presenza di eventuali forti correlazioni tra le stesse.
- Supponiamo ad esempio di voler classificare le persone sulla base dell'altezza e della lunghezza del piede. Ogni pattern x (bidimensionale) risulta costituito da due feature (x_1 = altezza, x_2 = lunghezza del piede).

52

Classificatore Nearest Neighbour (NN)- caso studio



Lo spazio di variazione dell'altezza (210-140 = 70 cm) risulta molto maggiore di quello della lunghezza del piede (40-20=20 cm). Pertanto se la similarità tra pattern venisse misurata con semplice distanza euclidea la componente altezza "peserebbe" molto di più della componente lunghezza del piede.
Per evitare i problemi legati a diversi spazi di variazioni delle feature, ogni feature i-esima dovrebbe essere normalizzata per il relativo spazio di variazione v_i. (si veda normalizzazione delle feature)
Gli spazi di variazione v_i, i=1..d possono essere derivati dal Training Set, come differenza tra massimo e minimo valore per la feature i-esima, o meglio, come massimo meno minimo dopo aver rimosso il 2...5% dei valori più alti e più bassi (per escludere outlier).

53

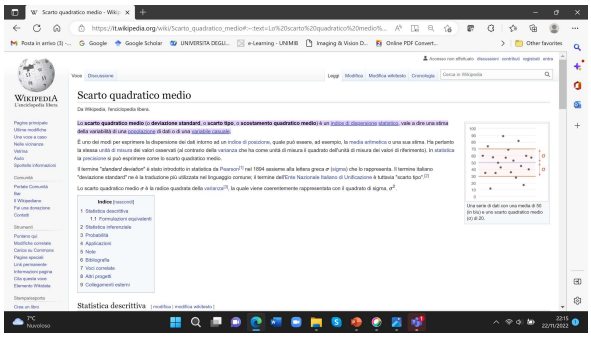
Classificatore Nearest Neighbour (NN)

Volendo è anche possibile esplicitare pesi *pi* diversi per le diverse feature. I pesi *pi*, i=1..*d* (anch'essi derivati dal Training Set) possono essere scelti proporzionalmente al ***potere discriminante*** delle feature calcolabile ad esempio come rapporto:

$$p_i = \frac{variabilità - interclasse_i}{variabilità - intraclasse_i}$$

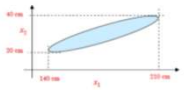
Per ***variabilità-intraclasse*** ci si riferisce alla variabilità della feature i-esima nell'ambito di ciascuna classe, e può essere calcolata come media degli scarti quadratici dei valori di x_i all'interno di ciascuna classe.
Per ***variabilità-interclasse*** ci si riferisce alla variabilità della feature i-esima per classi diverse. Può essere calcolata come scarto quadratico dei valori di x_i di un egual numero di campioni presi da ciascuna classe.

54



55

Classificatore Nearest Neighbour (NN)



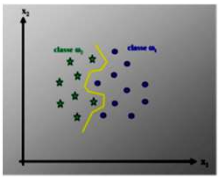
$$L_k(\mathbf{a},\mathbf{b}) = \left(\sum_{i=1}^{d} |a_i - b_i|^k\right)^{1/k} \quad \rightarrow \quad L_k(\mathbf{a},\mathbf{b}) = \left(\sum_{i=1}^{d} p_i \cdot \frac{|a_i - b_i|}{v_i}\right)^{k}^{1/k}$$

RECAP: Per evitare i problemi legati a diversi spazi di variazioni delle feature, ogni feature i-esima è normalizzata per il relativo spazio di variazione v_i. Volendo è anche possibile esplicitare pesi p_i diversi per le diverse feature

56

Classificatore Nearest Neighbour (NN)

E' semplice, non ha bisogno di addestramento; è applicabile a qualsiasi tipo di distribuzione statistica, ha accuratezza elevata (se k è sufficientemente grande). Ma, ha un onere computazionale legato alla fase di classificazione è elevato se in numero di campioni è elevato: spesso si ricorre a tecniche di partizione dello spazio (ad esempio il Kd-tree) per accelerare la ricerca dei k punti vicini.



Le frontiere di decisione prodotte dal k-NN sono di tipo lineare a tratti.

57

Classificazione con rigetto

Ci possono essere casi in cui il costo di un errore è così elevato che è conveniente astenersi dal fornire una risposta piuttosto che rischiare un errore. In questi casi, alle decisioni possibili si aggiunge la "decisione di non decidere", detta anche *rigetto*.

Le condizioni per le quali viene sospesa la decisione vanno sotto il nome di *regola di rigetto* (*reject rule*).

58

Classificatore bayesiano

- Sia V uno spazio di pattern d-dimensionali e W = {w_1, w_2, ..., w_s} un insieme di classi disgiunte costituite da elementi di V
- Per ogni x∈V e per ogni w_i∈W, indichiamo con $p(x|w_i)$ la *densità di probabilità condizionale* di x data w_i, ovvero la densità di probabilità che il prossimo pattern sia x sotto l'ipotesi che la sua classe di appartenenza sia w_i
- Per ogni w_i∈W, indichiamo con $P(w_i)$ la *probabilità a priori* di w_i ovvero la probabilità, indipendentemente dall'osservazione, che il prossimo pattern da classificare sia di classe w_i
- Per ogni **x**∈**V** indichiamo con $p(\mathbf{x})$ la *densità di probabilità assoluta* di **x**, ovvero la densità di probabilità che il prossimo pattern da classificare sia **x**.

$$p(\mathbf{x}) = \sum_{i=1}^{s} p(\mathbf{x}|w_i) \cdot P(w_i) \qquad \text{dove} \qquad \sum_{i=1}^{s} P(w_i) = 1$$

59

Classificatore bayesiano

Per ogni w_i∈W e per ogni **x**∈**V** indichiamo con $P(w_i|\mathbf{x})$ la ***probabilità a posteriori*** di w_i dato **x**, ovvero la probabilità che avendo osservato il pattern **x**, la classe di appartenenza sia w_i. Per il teorema di Bayes:

$$P(w_i|\mathbf{x}) = \frac{p(\mathbf{x}|w_i) \cdot P(w_i)}{p(\mathbf{x})}$$

Quindi, dato un pattern **x** da classificare in una delle s classi $w_1, w_2, \ldots, w_s$ di cui sono note:
- le probabilità a priori $P(w_1)$, $P(w_2)$, ... $P(w_s)$
- le densità di probabilità condizionali $p(\mathbf{x}|w_1)$, $p(\mathbf{x}|w_2)$, ... $p(\mathbf{x}|w_s)$

la ***regola di classificazione di Bayes*** assegna **x** alla classe b tale che:

$$P(w_b|\mathbf{x}) = \max_{i=1..s} \{P(w_i|\mathbf{x})\}$$

60

Metodo della Massima Verosimiglianza

- La conoscenza esatta delle probabilità a priori, e delle densità condizionali è possibile "solo in teoria"; pertanto nella pratica si fanno spesso ipotesi sulla forma delle distribuzioni
- Il metodo della Massima Verosimiglianza ("*Maximum Likelihood*", ML) sfrutta le caratteristiche statistiche delle classi fino al secondo ordine.
  - Si assume che la densità di probabilità delle classi sia del tipo gaussiano multidimensionale. Per molti applicazioni, questa assunzione rappresenta una buona approssimazione.
  - Di conseguenza, per ciascuna classe, è necessario calcolare sulla base dei campioni di training:
    - Vettore medio (baricentro);
    - Matrice di covarianza;

61

Distribuzione normale

$$p(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}\cdot|\Sigma|^{1/2}}\cdot exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^t\,\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right]$$

$$\boldsymbol{\mu} = [\mu_1, \mu_2, \ldots \mu_d]^t \quad \mu_i = E[x_i]$$

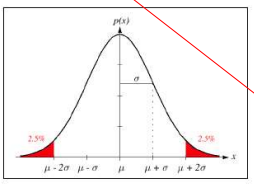
$$\Sigma = [\sigma_{ij}] \quad \sigma_{ij} = E[(x_i-\mu_i)(x_j-\mu_j)]$$

dove **x** è un vettore colonna *d*-dimensionale, μ è il vettore media della distribuzione, è la matrice di covarianza (*d*×*d*), $|\Sigma|$ e Σ^{-1} sono rispettivamente il determinante e l'inversa di Σ ; *E*[.] indica il valore atteso (expected) di una variabile aleatoria.

Se la matrice di covarianza è diagonale, la distribuzione normale multidimensionale è definita come semplice prodotto di d Normali monodimensionali. In tal caso gli assi principali sono paralleli agli assi cartesiani.

62

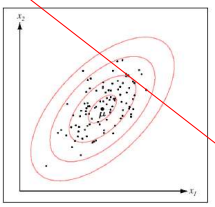
Distribuzione normale



Nel caso 1-dimensionale, la distribuzione normale è controllata dal valor medio μ e dallo scarto σ.
Solo il 5% circa del "volume" è esterno all'intervallo $[\mu-2\sigma, \mu+2\sigma]$.
Solitamente si assume che la distribuzione valga 0 a distanze maggiori di 3σ dal valore medio.

63

Distribuzione normale



Nel caso d-dimensionale, la distribuzione normale è controllata dal vettore medio μ (*d* valori) e dalla matrice di covarianza Σ (*d* (*d*+1)/2 valori indipendenti).
La generazione di campioni con distribuzione multinormale forma una nuvola di punti (iper-ellissoide), il cui centro coincide con μ, e la forma dalla matrice di covarianza è determinata da Σ.
Il luogo dei punti con densità costante sono gli iper-ellissoidi per cui la forma quadratica

$$(\mathbf{x}-\boldsymbol{\mu})^t\,\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

è costante. Gli assi dell'iper-ellissoide sono definiti dagli autovettori di Σ.

Distanza di Mahalanobis

La distanza di Mahalanobis *r* tra **x** e μ, definita dall'equazione:

$$r^2 = (\mathbf{x}-\boldsymbol{\mu})^t\,\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})$$

64

Metodo della Massima Verosimiglianza

- Misure in N-dimensioni, nota covarianza C_j e medie $\mathbf{m}_j$ della popolazione
- La densità di probabilità è la Gaussiana N-D

$$p(\vec{x}\,|\,Y_j) = \frac{1}{(2\pi)^{N/2}|C_j|^{1/2}}\exp\left[-\frac{1}{2}(\vec{x}-\vec{m}_j)^t\,C_j^{-1}(\vec{x}-\vec{m}_j)\right]$$

- Devo quindi trovare j per cui è massimo $D_j(\mathbf{x}) = p(\mathbf{x}\,|\,Y_j)P(Y_j)$

65

Classificazione con rigetto

Ci possono essere casi in cui il costo di un errore è così elevato che è conveniente astenersi dal fornire una risposta piuttosto che rischiare un errore.
In questi casi, alle decisioni possibili si aggiunge la "decisione di non decidere", detta anche *rigetto*.

Le condizioni per le quali viene sospesa la decisione vanno sotto il nome di *regola di rigetto* (*reject rule*).

Per il classificatore bayesiano, la probabilità di errore su un campione x è $P_e(x) = 1-\max\{P(\omega_i\,|\,x)\}$. Supponiamo di non voler procedere alla classificazione se la Pe supera una soglia t (P_e massima tollerabile).

66

Multi-classificatore

- Diversi classificatori possono essere utilizzati (*normalmente in parallelo, ma talvolta anche in cascata o in modo gerarchico*) per eseguire la classificazione dei pattern; le decisioni dei singoli classificatori sono ***fuse*** ad un qualche livello della catena di classificazione.
- La combinazione è ***efficace*** solo nel caso in cui i singoli classificatori siano in qualche modo ***indipendenti tra loro***, ovvero non commettano tutti lo stesso tipo di errori.
- L'indipendenza (o diversità) è normalmente ottenuta cercando di:
  - Utilizzare feature diverse (e.g. colore e texture)
  - Utilizzare algoritmi diversi per l'estrazione delle feature (e.g. RGB, HSI, ….)
  - Utilizzare diversi algoritmi di classificazione
  - Addestrare lo stesso algoritmo di classificazione su training set diversi (***bagging***)
  - Insistere nell'addestramento di alcuni classificatori con i pattern più frequentemente erroneamente classificati (***boosting***)
- La combinazione può essere eseguita a ***livello di decisione*** o a ***livello di confidenza***.

67

Multi-classificatore

Fusione a livello di decisione

- Ogni singolo classificatore fornisce in output la propria decisione che consiste della classe cui ha assegnato il pattern e opzionalmente del livello di affidabilità della classificazione eseguita (*ovvero di quanto il classificatore si sente sicuro della decisione presa*).
- Le decisioni possono essere tra loro combinate in diversi modi. Uno dei più noti e semplici metodi di fusione è la cosiddetta ***majority vote rule***; ogni classificatore vota per una classe, il pattern viene assegnato alla classe maggiormente votata.

68

Multi-classificatore

Fusione a livello di confidenza

- Ogni singolo classificatore C_j, j=1..NC fornisce in output la confidenza di classificazione del pattern rispetto a ciascuna delle classi, ovvero un vettore $\mathbf{conf}_j=[conf_{j1}, conf_{j2},\ldots conf_{js}]$ di dimensionalità s in cui l'i-esimo elemento indica la probabilità di appartenenza del pattern alla classe i-esima.
- Diversi metodi di fusione sono possibili tra cui (somma, media, prodotto, max, min).
- Bisogna prestare attenzione alla normalizzazione dei vettori di confidenza nel caso in cui essi non siano probabilità (ma ad esempio similarità). Infatti in quest'ultimo caso i valori non sono tra loro confrontabili e una fase di normalizzazione è necessaria.

69

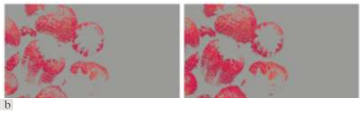
Binarizzazione mediante classificazione

70

Binarizzazione mediante classificazione

$$s_i = \begin{cases} 0.5 & \text{if } \left[|r_j - a_j| > \frac{W}{2}\right]_{any\ 1\le j\le n} \\ r_i & \text{otherwise} \end{cases}, \quad i = 1, 2, \ldots, n.$$

$$s_i = \begin{cases} 0.5 & \text{if } \sum_{j=1}^{n} (r_j - a_j)^2 > R_0^2 \\ r_i & \text{otherwise} \end{cases}, \quad i = 1, 2, \ldots, n.$$



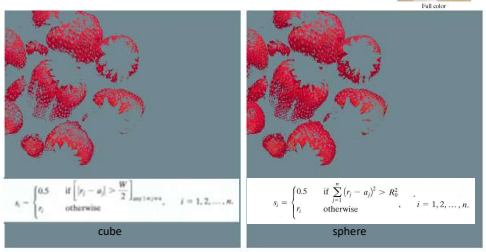
a b

FIGURE 6.34 Color-slicing transformations that detect (a) reds within an RGB cube of width $W = 0.2549$ centered at $(0.6863, 0.1608, 0.1922)$, and (b) reds within an RGB sphere of radius 0.1765 centered at the same point. Pixels outside the cube and sphere were replaced by color $(0.5, 0.5, 0.5)$.

71

Binarizzazione mediante classificazione

Full color



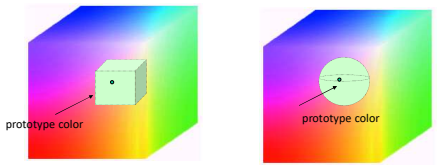
cube

sphere

Come posso trovare il miglior valore per W o R?

72

Binarizzazione mediante classificazione



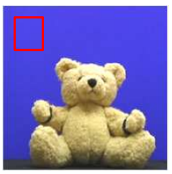
- I colori di interesse possono essere racchiusi da cubi (ipercubo) di larghezza W e centrati a (a1, a2, ..., an)
- I colori di interesse possono essere racchiusi da sfere (iperspere) di raggio R0 e centrati a (a1, a2, ..., an)

- Come posso definire il centroide (colore atteso) e il suo intorno?

73

Binarizzazione mediante classificazione

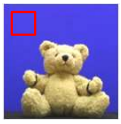
Il colore atteso μ, viene tipicamente stimato a partire da una (o più) immagini di *training*. Interpretando quindi il colore di un pixel dell'oggetto come una variabile aleatoria a tre dimensioni, il colore atteso è ottenuto stimandone il *valor medio* a partire dai *training sample* disponibili.



74

Binarizzazione mediante classificazione

Denotando quindi il colore di un pixel come: I (p) la segmentazione di un'immagine può essere ottenuta calcolando per ogni pixel la distanza (e.g. euclidea) rispetto al colore atteso (μ) dell'oggetto di interesse e marcando come *sfondo* i pixel per i quali tale distanza è inferiore ad una soglia (intorno):



$$\forall p \in \mathbf{I}: \begin{cases} d(\mathbf{I}(p),\boldsymbol{\mu}) \le T \rightarrow O(p) = F \\ d(\mathbf{I}(p),\boldsymbol{\mu}) > T \rightarrow O(p) = B \end{cases}$$

$$d(\mathbf{I}(p),\boldsymbol{\mu}) = \left((I_r(p)-\mu_r)^2 + (I_g(p)-\mu_g)^2 + (I_b(p)-\mu_b)^2 \right)^{\frac{1}{2}}$$

Come posso definire T, la soglia, ovvero l'intorno?

75

Binarizzazione mediante classificazione

- Data un'immagine a colori RGB I = (I_R, I_G, I_B) con una sotto-regione R_k, possiamo calcolare la media locale e la varianza per ogni canale di colore come

$$\boldsymbol{\mu}_k(\boldsymbol{I},u,v) = \begin{pmatrix} \mu_k(I_R,u,v) \\ \mu_k(I_G,u,v) \\ \mu_k(I_B,u,v) \end{pmatrix}, \quad \boldsymbol{\sigma}^2_k(\boldsymbol{I},u,v) = \begin{pmatrix} \sigma^2_k(I_R,u,v) \\ \sigma^2_k(I_G,u,v) \\ \sigma^2_k(I_B,u,v) \end{pmatrix},$$

- La varianza complessiva σ^2_k può essere definita in modi diversi, ad esempio, come la somma delle varianze nei singoli canali di colore, cioè

$$\sigma^2_{k,\mathrm{RGB}}(\boldsymbol{I},u,v) = \sigma^2_k(I_R,u,v) + \sigma^2_k(I_G,u,v) + \sigma^2_k(I_B,u,v).$$

76

Binarizzazione mediante classificazione

In alternativa si potrebbe definire la varianza di colore combinata come la norma della matrice di covarianza del colore 3x3 per la sotto-regione R_k,

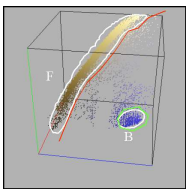
$$\Sigma_k(\boldsymbol{I},u,v) = \begin{pmatrix} \sigma_{k,\mathrm{RR}} & \sigma_{k,\mathrm{RG}} & \sigma_{k,\mathrm{RB}} \\ \sigma_{k,\mathrm{GR}} & \sigma_{k,\mathrm{GG}} & \sigma_{k,\mathrm{GB}} \\ \sigma_{k,\mathrm{BR}} & \sigma_{k,\mathrm{BG}} & \sigma_{k,\mathrm{BB}} \end{pmatrix},$$

with $$\sigma_{k,pq} = \frac{1}{|R_k|} \cdot \sum_{(i,j)\in R_k} \big[I_p(u+i,v+j) - \mu_k(I_p,u,v)\big] \cdot \big[I_q(u+i,v+j) - \mu_k(I_q,u,v)\big],$$

$$\sigma^2_{k,\mathrm{RGB}} = \|\Sigma_k(\boldsymbol{I},u,v)\|^2_2 = \sum_{\substack{p,q\in \\ \{\mathrm{R},\mathrm{G},\mathrm{B}\}}} (\sigma_{k,pq})^2$$

77

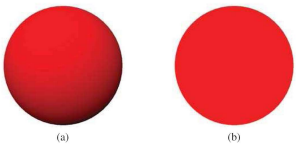
Binarizzazione mediante classificazione



Il metodo descritto funziona per la classe B che è compatta limitata ad una solo porzione dello spazio RGB e quasi sferica. Avrei potuto usare questo metodo per caratterizzare la classe F nello spazio RGB?

78

Recap - spazi colore



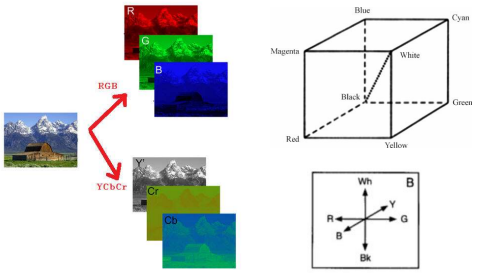
(a) (b)

Figure 5.4 Sphere illuminated by a single light source (a). Segmentation is simpler if only hue is considered (b).

- Non sempre e' utile usare lo spazio colore RGB
- Non sempre e' utile usare tutte le dimensioni di una spazio colore

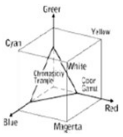
79

Recap - spazi colore

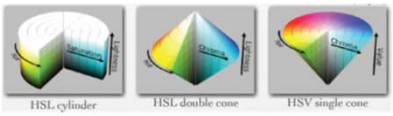


80

Recap - spazi colore



$$\begin{aligned} \text{intensity } I &= (R+G+B)/3 \\ \text{normalized red } r &= R/(R+G+B) \\ \text{normalized green } g &= G/(R+G+B) \\ \text{normalized blue } b &= B/(R+G+B) \end{aligned}$$



81

HSI

$$\text{I} > 40 \text{ and } \begin{cases} 13<S<110 \text{ and } 0°<H<28° \\ \text{or} \\ 13<S<110 \text{ and } 332°<H<360° \\ \text{or} \\ 13<S<75 \text{ and } 309°<H<331° \end{cases}$$

$$I_1 = \frac{1}{3}(R+G+B); \quad I_2 = \frac{1}{2}(R-B); \quad I_3 = \frac{1}{4}(2R-R-B)$$

$$I = I_1; \quad S = \sqrt{I_2^2 + I_3^2} \quad H = \tan^{-1}\left(\frac{I_3}{I_2}\right)$$

I-S. Hsieh, K-C. Fan, and C. Lin, "A statistic approach to the detection of human faces in colour nature scene", Pattern Recognition 35, 1583-1596 (2002).

82

HSI

$$\text{I} > 40 \text{ and } \begin{cases} 13<S<110 \text{ and } 0°<H<28° \\ \text{or} \\ 13<S<110 \text{ and } 332°<H<360° \\ \text{or} \\ 13<S<75 \text{ and } 309°<H<331° \end{cases}$$

$$I_1 = \frac{1}{3}(R+G+B); \quad I_2 = \frac{1}{2}(R-B); \quad I_3 = \frac{1}{4}(2R-R-B)$$

$$I = I_1; \quad S = \sqrt{I_2^2 + I_3^2} \quad H = \tan^{-1}\left(\frac{I_3}{I_2}\right)$$

I-S. Hsieh, K-C. Fan, and C. Lin, "A statistic approach to the detection of human faces in colour nature scene", Pattern Recognition 35, 1583-1596 (2002).

83

RGB

Uniform daylight illumination

R > 95 and G > 40 and B > 20

and

Max {R,G,B} – min {R,G,B} < 15

and |R-G| > 15 and R > G and R > B

Flashlight or daylight lateral illumination

R > 220 and G > 210 and B > 170

and

|R-G| ≤ 15 and B < R and B < G

J. Kovac, P. Peer and F. Solina, "2D versus 3D colour space face detection", 4th EURASIP Conference on Video/Image Processing and Multimedia Communications (Croatia, 2003), pp. 449-454.

84

YCbCr1

77 <= Cb <=127

and

133 <= Cr <= 173

D. Chai and K. N. Ngan, "Face segmentation using skin colour map in videophone applications", IEEE Transactions on Circuits and Systems for Video Technology 9, 551-564 (1999).

85

YCbCr2 (a)

Cb Cr as function of Y → transformed chroma Cb'(Y) Cr'(Y)

$$C'_{i=b,r}(Y) = \begin{cases} (C_i(Y) - \overline{C_i}(Y)) \cdot \frac{Wc_i}{Wc_i(Y)} + \overline{C_i}(K_h) & \text{if } Y < K_l \text{ or } K_h < Y \\ C_i(Y) & \text{if } Y \in [K_l, K_h] \end{cases}$$

$$\overline{C_r}(Y) = \begin{cases} 154 + \frac{(K_l - Y)\cdot(154-144)}{K_l - Y_{\min}} & \text{if } Y < K_l \\ 154 + \frac{(Y - K_h)\cdot(154-132)}{Y_{\max} - K_h} & \text{if } K_h < Y \end{cases} \qquad \overline{C_b}(Y) = \begin{cases} 108 + \frac{(K_l - Y)\cdot(118-108)}{K_l - Y_{\min}} & \text{if } Y < K_l \\ 108 + \frac{(Y - K_h)\cdot(118-108)}{Y_{\max} - K_h} & \text{if } K_h < Y \end{cases}$$

$$Wc_i(Y) = \begin{cases} WLc_i + \frac{(Y - Y_{\min})\cdot(WC_i - WLc_i)}{K_l - Y_{\min}} & \text{if } Y < K_l \\ WHc_i + \frac{(Y_{\max} - Y)\cdot(WC_i - WHc_i)}{Y_{\max} - K_h} & \text{if } K_h < Y \end{cases}$$

K_l=125; K_h=188; WC_b = 46,97; WC_r = 38,76; WLc_b = 23; WLc_r = 20; WHc_b = 14; WHc_r = 10.

86

YCbCr2 (b)

The skin cluster is described in the transformed Cb',Cr' with the ellipse:

$$\frac{(x - ec_x)^2}{a^2} + \frac{(y - ec_y)^2}{b^2} = 1$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} Cb' - c_x \\ Cr' - c_y \end{bmatrix}$$

ec_x = 1.60; ec_y = 2.41;
a = 25.39; b =14,03;
c_x = 109,38 ; c_y = 152,02;
θ = 2.53 rad;

R. Hsu, M. Abdel-Mottaleb and A. K. Jain, "Face detection in colour images", IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 696-706 (2002).

87

HSV1

V ≥ 0.4

and

0° ≤ H ≤ 25° or 335° ≤ H ≤ 360°

and

0.2 ≤ S ≤ 0.6

S. Tsekeridou and I. Pitas, "Facial feature extraction in frontal views using biometric analogies", Proc. of the IX European Signal Processing Conference, vol. I, 315-318 (1998).

88

HSV2

S>=10 and V>=40
and
H<=(-0.4*V+75)
and
S<=(-H-0.1*V+110)

and

H>=0 and S<=(0.08*(100-V).*H+0.5*V)
or
(H<0) and (S<=(0.5*H+35));

NB. H ∈ [-180° 180°]

C.Garcia and G. Tziritas, "Face detection using quantized skin colour regions merging and wavelet packet analysis", IEEE Transaction on Multimedia 1, 264-277 (1999).

89

rgb

$$\frac{r}{g} > 1.185$$

and

$$\frac{r*b}{(r+g+b)^2} > 0.107$$

and

$$\frac{r*g}{(r+g+b)^2} > 0.112$$

$$r = \frac{R}{R+G+B}; \quad g = \frac{G}{R+G+B}; \quad b = \frac{B}{R+G+B}$$

G. Gomez and E. F. Morales, "Automatic feature construction and a simple rule induction algorithm for skin detection", Proc. Of the ICML workshop on Machine Learning in Computer Vision, A. Sowrnya, T. Zrimec (eds), 31-38 (2002).

90

91

Valutazione quantitativa

92

Valutazione quantitativa

- Non si puo' fare senza dati !
- La valutazione quantitativa fornisce risultati numerici, dal confronto tra il risultati della classificazione fornita ed un sottoinsieme della realtà che prende il nome di INSIEME DI VERIFICA (*test set*)
- Le questioni principali da affrontare prima di intraprendere una valutazione quantitativa sono:
  - *la scelta dell'insieme di verifica;*
  - *la dimensione dell'insieme di verifica.*
- Training e test set non dovrebbe sovrapporsi, neanche parzialmente, con l'insieme di addestramento, altrimenti si falsano le accuratezze in senso ottimistico.
- La dimensione del campione utilizzato deve essere sufficientemente grande da risultare statisticamente rappresentativo
- Spesso si effettua un unico campionamento per poi distribuire i campioni tra i due insiemi.

93

Come si valuta una classificazione binaria


esempio di classificazione binaria: skin detector

2 classi : skin , no skin

True positive

False positive

False negative


true positive (TP) : n° di pixel correttamente assegnati alla classe di skin;
false positive (FP) : n° di pixel non-skin assegnati in maniera errata alla classe skin.
false negative (FN) : n° di pixel skin assegnati in maniera errata alla classe non-skin.

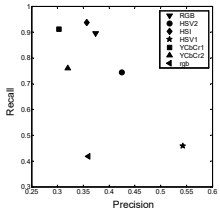
94

come si valuta un classificatore binario

Recall=*TP/(TP+FN)*

Precision= *TP/(TP+FP)*

In genere al crescere dalla precisione diminuisce la recall e viceversa



Se al sistema è concesso di rigettare pattern ovvero di non-classificarli o riconoscerli in caso di elevata incertezza, e necessario misurare le prestazioni (errori di classificazione) in funzione della percentuale di reiezione concessa

95

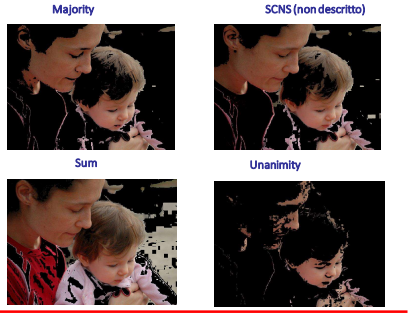
Combinazione dei classificatori

Voting methods

1. sum rule $C_{sum} = \bigcup_{i=1}^{N} C_i$
2. unanimity $C_{product} = \bigcap_{i=1}^{N} C_i$
3. majority $C_{majority} = \left(\sum_{i=1}^{N} C_i \right) \geq N/2$

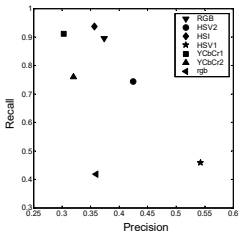
96

Combinazione dei classificatori



97

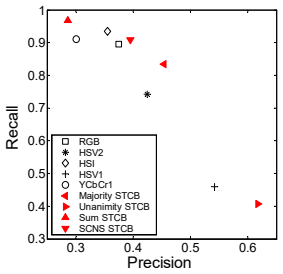
Valutazione dei singoli algoritmi



- Quale e' l'algoritmo migliore ?

98

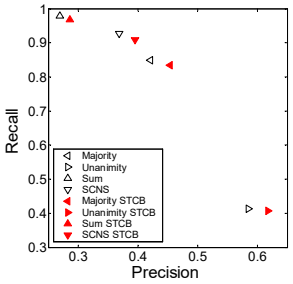
Valutazione delle possibili combinazioni



Idee per migliorare ?

99

Preprocessing (white balancing)
Valutazione delle possibili combinazioni



100

Come si valuta un classificatore a n classi

Consideriamo una classificazione in cui ogni caso è assegnato a una fra *k* classi predefinite.

Dal punto di vista conoscitivo la misura ovvia per una classificazione è la percentuale di casi ben classificati, oppure, vista al contrario, la percentuale di errori di classificazione. (Se le classi sono due, possiamo chiamarle *Positivi* e *Negativi* e ragionare in termini di costo dei falsi positivi e dei falsi negativi).

101

Esempio di matrice di confusione

		Classi di realtà a terra		
		Acqua	Vegetazione	Urbano
Classi assegnate	Acqua	1345	73	84
	Vegetazione	62	2315	37
	Urbano	123	49	678

- Sulla diagonale principale si trovano gli elementi correttamente classificati. Gli elementi fuori diagonale rappresentano errori di classificazione:
  - **di omissione** (*omission error*), quando un pixel appartenente alla classe considerata non vi è inserito;
  - **di inclusione** (*commission error*) quando un pixel è assegnato alla classe considerata pur non appartenendovi.

102

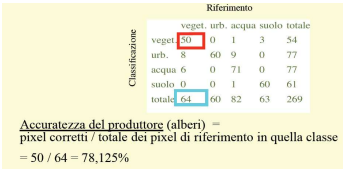
Esempio Matrice di confusione

- L'accuratezza totale dà una misura complessiva di quanto la classificazione è stata ben fatta, ma non distingue tra gli errori commessi nelle diverse classi, che sono trattate tutte allo stesso modo.
- A volte però si è specificamente interessati ad una classe in particolare, perciò sarebbe opportuno definire indici di accuratezza legati ad una specifica classe.
- Ad esempio, il destinatario della mappa di classificazione potrebbe essere interessato a sapere quanto si può fidare del fatto che un pixel assegnato alla classe **vegetazione** sia effettivamente appartenente a quella classe.

103

Matrice di confusione

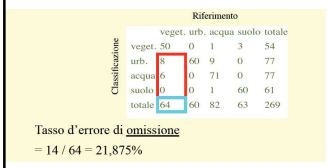
- L'accuratezza per l'utente è definita come il rapporto tra il numero di pixel correttamente classificati nella classe considerata ed il numero di pixel assegnati in totale a quella classe.
- Analogamente si può definire l'accuratezza per il produttore (o del produttore) che misura quanto dell'insieme di verifica pertinente ad una determinata classe è stato effettivamente assegnato a quella classe.

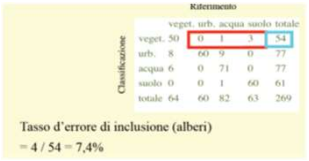


104

Esempio Matrice di confusione

Altri modi, sostanzialmente analoghi, di valutare la "bontà" della classificazione distinguendo tra classi è quella dei così detti tassi di errore di omissione e di inclusione.





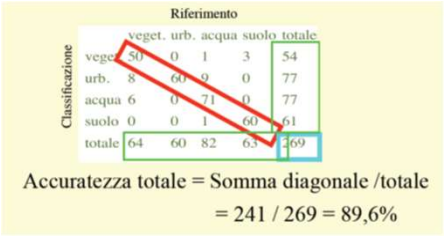
105

Accuratezze totale

- L'accuratezza per l'utente è definita come il rapporto tra il numero di pixel correttamente classificati nella classe considerata ed il numero di pixel assegnati in totale a quella classe.
- Analogamente si può definire l'accuratezza per il produttore (o del produttore) che misura quanto dell'insieme di verifica pertinente ad una determinata classe è stato effettivamente assegnato a quella classe.
- Il fatto che i pixel classificati correttamente si trovino tutti e soli sulla diagonale principale suggerisce un modo per valutare la bontà della classificazione.
- Si può infatti definire la così detta *accuratezza totale* (*overall accuracy*) come il rapporto fra il numero totale di pixel correttamente classificati (Σ dei contenuti degli elementi sulla diagonale principale o traccia della matrice) ed il numero totale di pixel considerati nell'insieme di verifica.

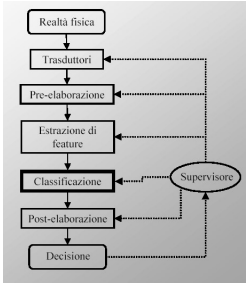
106

Accuratezze totale



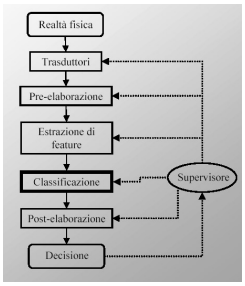
107

Progetto di applicazioni di classificazione



108

Progetto di applicazioni di classificazione



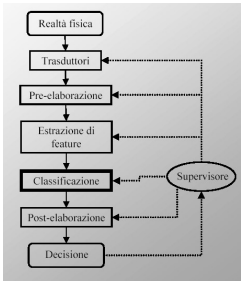
Acquisizione

L'ingresso di un sistema di riconoscimento viene spesso alimentato da uno o più trasduttori (telecamera, microfono,...).

Le caratteristiche e limitazioni del trasduttore (banda passante, risoluzione, distorsione, rapporto segnale/rumore) giocano un ruolo importante nella progettazione e sulle prestazioni di un sistema di riconoscimento.

109

Progetto di applicazioni di classificazione



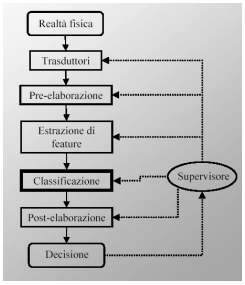
Pre-elaborazione

In questa fase, i dati acquisiti subiscono alcune operazioni finalizzate a facilitare il processo di riconoscimento (ad esempio, filtraggio, correzioni radiometriche e/o geometriche ecc.).

I pattern vengono eventualmente isolati l'uno dall'altro mediante un processo di segmentazione

110

Progetto di applicazioni di classificazione

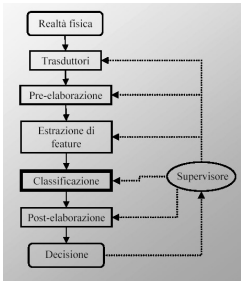


Estrazione di Feature

Lo scopo di un estrattore di feature è di trovare misure e/o caratteristiche che permettano di discriminare al meglio oggetti differenti e che siano (se richiesto dall'applicazione) insensibili a problemi di traslazione, di rotazione oppure di scala.

111

Progetto di applicazioni di classificazione



Classificazione

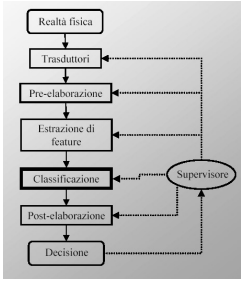
è il processo decisionale vero e proprio

Post-elaborazione

Permette di sfruttare i risultati forniti dal/i classificatore/i per migliorare le prestazioni della classificazione usando ad esempio l'informazione contestuale spaziale

112

Progetto di applicazioni di classificazione

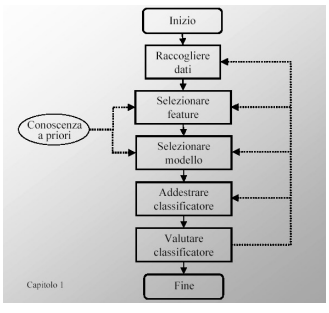


Supervisore

Alla luce dei risultati ottenuti, il supervisore decide di apportare una o più modifiche al sistema di riconoscimento in funzione dell'applicazione scelta (ad es., nuova regolazione dei parametri di acquisizione di una telecamera, cambiamento dell'insieme delle feature, cambiamento della regola di classificazione);

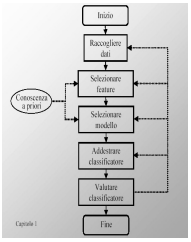
113

Progetto di applicazioni di classificazione



114

Progetto di applicazioni di classificazione

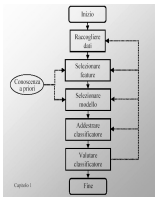


Raccolta dei Dati

Come sapere se è stato raccolto un insieme di campioni di training (per allenare il classificatore) o di test (per testare il classificatore) sufficientemente grande e rappresentativo?

115

Progetto di applicazioni di classificazione



Per guidare la scelta delle feature da usare per descrivere un oggetto, occorre evidenziare alcune proprietà che queste dovrebbero soddisfare:

- assumere valori significativamente diversi per oggetti appartenenti a classi diverse (proprietà di discriminazione);
- assumere valori simili per oggetti appartenenti alla stessa classe (proprietà di affidabilità);
- risultare indipendenti l'una dall'altra (proprietà di indipendenza);
- essere in numero minimo (proprietà di minima cardinalità).
- essere insensibili al rumore di acquisizione

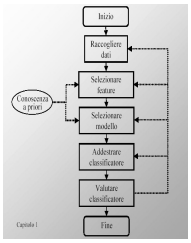
Possibili feature

-Numeri reali (continuo)
- Numeri interi (discreto)
-Booleani
- Etichette, categorie (ordinate, non ordinate)

Possono anche essere mischiate!

116

Progetto di applicazioni di classificazione



Selezione del Modello

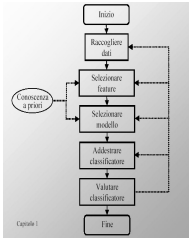
Il modello scelto per la classificazione può giocare un ruolo fondamentale nelle prestazioni ottenute. Maggiore è la complessità dei dati e dell' applicazione considerata, maggiore deve essere la complessità del classificatore per poter raggiungere risultati soddisfacenti.

Addestramento del Classificatore

L'addestramento consiste nell'utilizzare i campioni di training per determinare i parametri del modello del classificatore.

117

Progetto di applicazioni di classificazione



Parametri per la Valutazione di un Classificatore

Accuratezza: uno dei criteri più utilizzati nella valutazione dei risultati della classificazione dei campioni di test è l'accuratezza, cioè il numero di campioni correttamente classificati rispetto al numero totale di campioni classificati.

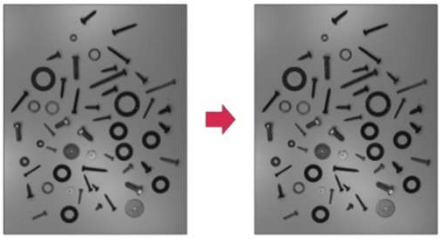
Onere Computazionale: un altro criterio importante è la complessità del sistema di riconoscimento e, quindi, il suo impatto sul tempo di elaborazione. Tale impatto può essere più o meno critico a secondo dell'applicazione considerata (applicazione off-line oppure in tempo reale).

118

Un esempio completo

119

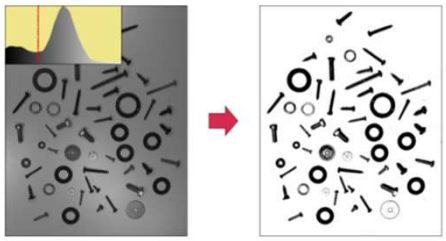
Un esempio completo – 1) Preprocessing



Applicazione di un filtro di sharpening per evidenziare maggiormente i contorni degli oggetti

120

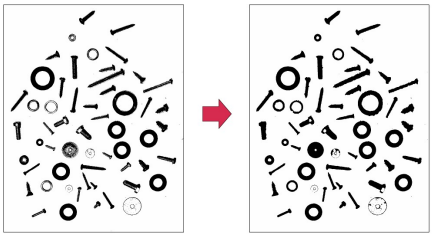
Un esempio completo – 2) Segmentazione



Binarizzazione con soglia determinata mediante analisi dell'istogramma

121

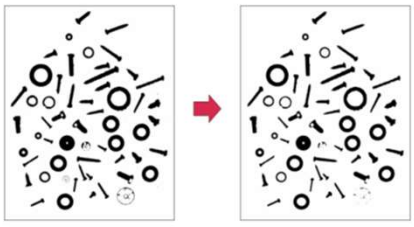
Un esempio completo – 3) Rimozione imperfezioni



Operatore morfologico di chiusura con elemento strutturante 7x7 (cerchio)

122

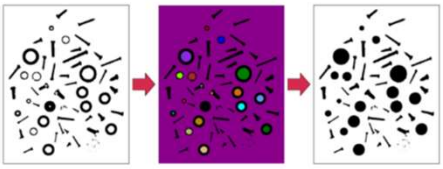
Un esempio completo – 4) Riduzione rumore



Operatore morfologico di erosione con elemento strutturante 3x3 (cerchio)

123

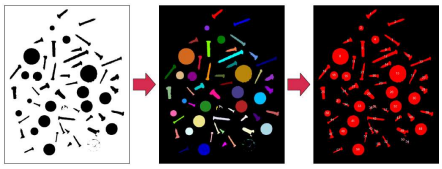
Un esempio completo – 5) Riempimento "buchi"



Etichettatura componenti connesse fatta sul background, quindi trasformazione in foreground di tutti i pixel appartenenti a "oggetti" con area inferiore ad una certa soglia (i "buchi")

124

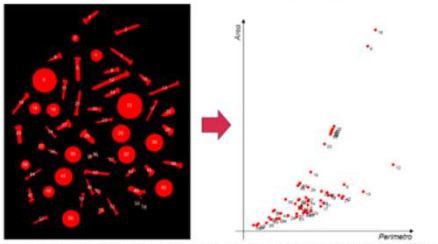
Un esempio completo – 6) Etichettatura oggetti



Etichettatura componenti connesse (sui pixel di foreground); solo gli oggetti con area maggiore di una certa soglia sono considerati

125

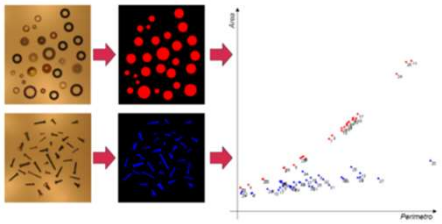
Un esempio completo – 7) Feature extraction



Due feature sono estratte da ogni oggetto etichettato: perimetro (numero di pixel confinanti con background) e area (numero totale di pixel dell'oggetto).

126

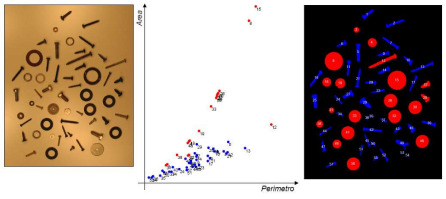
Un esempio completo – 8) Training set



Il training set viene costruito eseguendo gli stessi passi visti in precedenza su due immagini, ciascuna contenente solo oggetti di una classe

127

Un esempio completo – 9) Classificazione



Classificazione del test set utilizzando la semplice regola NN.
La maggior parte degli oggetti è correttamente classificata, ma vi sono alcuni errori evidenti: quali possono essere le cause?

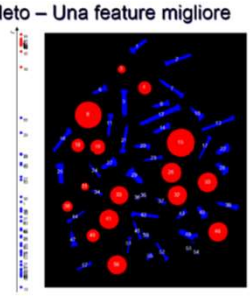
128

Un esempio completo – Una feature migliore

- Un miglioramento può essere ottenuto semplicemente utilizzando una feature *più discriminante*, derivata dalle due estratte:

$$r = \frac{area}{perimetro^2}$$

- La scelta delle feature è sempre una fase estremamente critica!
  - Non esistono feature migliori di altre a priori, la scelta va effettuata di volta in volta analizzando il problema specifico da risolvere.



129

Un esempio completo:

siete d'accordo o manca qualcosa ?

130

Un esempio quasi completo:
Per essere completo doveva includere una descrizione dei dati: training , test set….
E' fase di valutazione oggettiva

131

Introduzione alla classificazione

Raimondo Schettini
DISCo
Universita' di Milano Bicocca
schettini@disco.unimib.it
www.ivl.disco.unimib.it/Schettini/

132