

Classificazione di utensili

Introduzione

Descrizione del problema

Classificazione di tipi di utensili su sfondi vari

Obiettivo: classificare strumenti da lavoro.

Il training set comprende 10 classi di oggetti:

forbice, metro, pinza, chiave, martello, cacciavite, avvitatore, pappagallo, lima, pennarello.

5 tipi di sfondi: legno, cartone, gomma nera, piastrelle, uniforme bianco.

Dati

- Training Set: 91 immagini di un oggetto singolo
- Test Set: 27 immagini di più oggetti (con oggetti estranei)



Descrizione e analisi dei dati e assunzioni

ANALISI DEI DATI

- Gli sfondi potrebbero non essere omogenei
- Utensili della stessa classe possono avere diverse grandezze e colori
- Ombre e riflessi che possono creare problemi
- Alcuni utensili potrebbero cambiare forma (per esempio forbici aperte/chiuso)
- Possibili oggetti sconosciuti (non utensili)
- Immagini presenti hanno diverse risoluzioni e rapporto di forma (4/3 e 16/9)

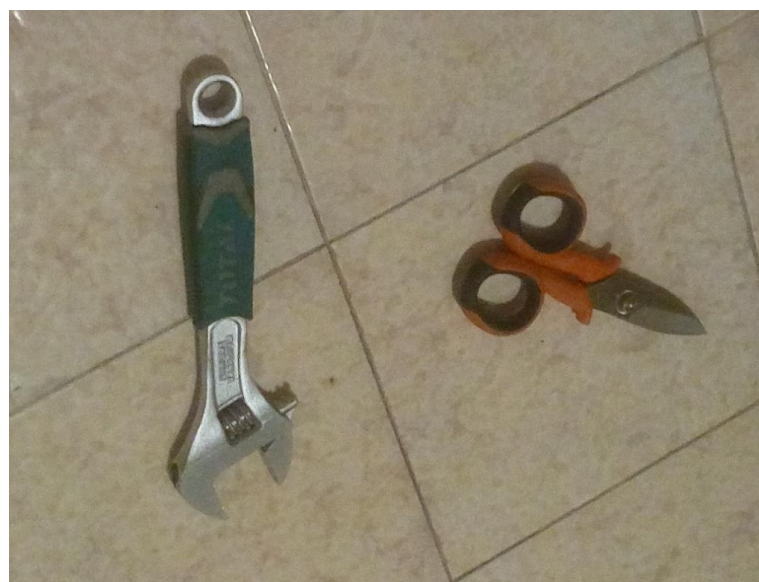
ASSUNZIONI

- Oggetti all'interno dei bordi senza toccarli
- Oggetti non si toccano fra loro
- Dieci possibili classi di utensili
- Immagini orientate orizzontalmente

Esempi foto del train set



Esempi foto del test set



Approccio di lavoro seguito

Scelte e motivazioni

Approccio di lavoro seguito

- Creazione di un dataset di immagini con un solo utensile su sfondo bianco
- Creazione delle immagini di ground truth annotata a mano (pixel-based)
- Prova di diverse tecniche di segmentazione per scegliere quella più adatta

Nota: noi elaboriamo tutte le immagini alla dimensione di 154x205 pixel.

Considerazioni: il thresholding funziona bene (solo) sulle immagini con sfondi uniformi, come pure kmeans sul valore di grigio dei pixel.

Approccio di lavoro seguito

- sauvola su finestra 30x30
- kmeans sul valore di grigio dei pixel.

raw



sauvola 30x30



kmeans



raw



sauvola 30x30



kmeans



Approccio di lavoro seguito

- sauvola su finestra 30x30
- kmeans sul valore di grigio dei pixel.

raw



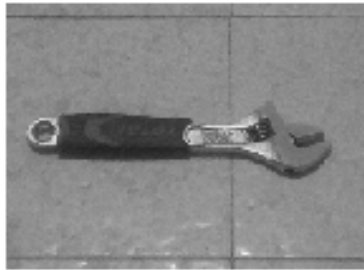
sauvola 30x30



kmeans



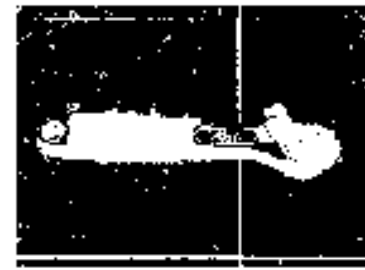
raw



sauvola 30x30



kmeans



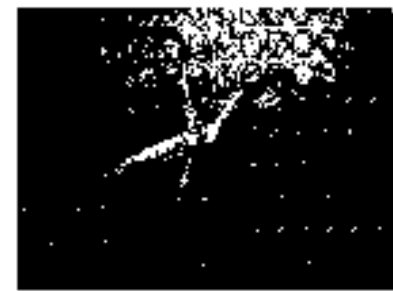
raw



sauvola 30x30



kmeans



Approccio di lavoro seguito

- Aggiunta di immagini su sfondi diversi(uno o più oggetti) e relative immagini annotate
- Prove con altre tecniche di segmentazione per scegliere quella più adatta su tutte le immagini

Prove fatte

- Filtraggio
- Binarizzazione Adattiva
- Gradiente
- Segmentazione con K-means (su colore e su lbp)
- Segmentazione con Morfologia
- Sharpening
- Trasformazione degli spazi colore
- Binarizzazione con Otsu
- Classificazione Sfondo con knn
- Diversi classificatori per lo sfondo
- Ecc...

(alcune) Prove fatte

raw



kmeans su lbp, dim 30x30, step 2



raw



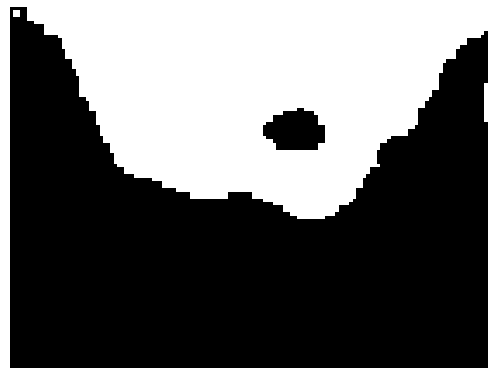
kmeans su lbp, dim 30x30, step2



raw



kmeans su lbp, dim 30x30, step2



Prove fatte

raw



sobel



raw



sobel

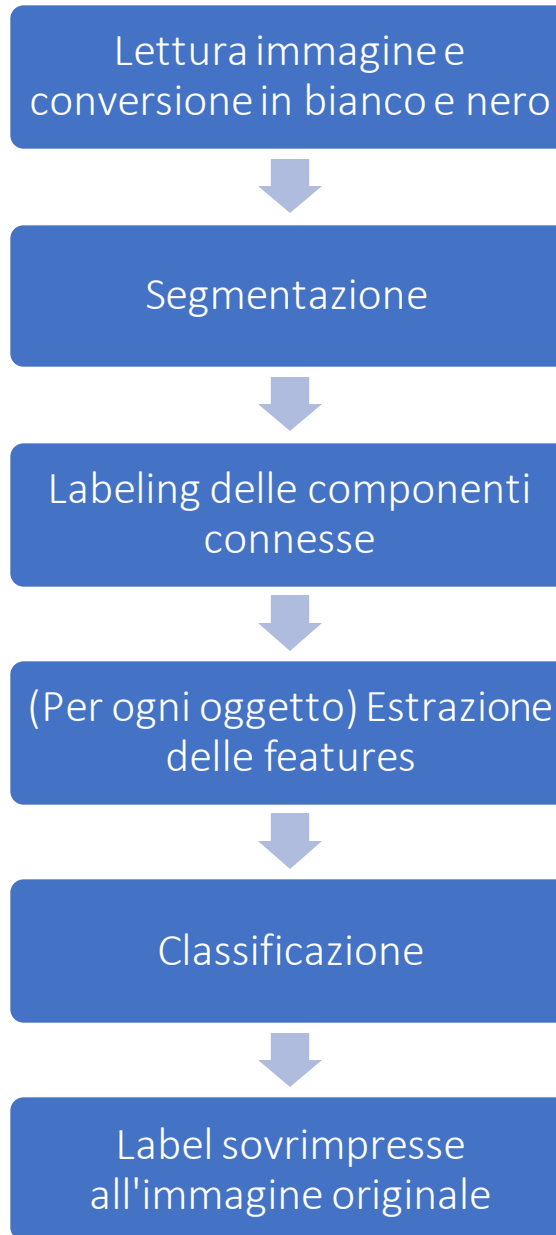


Approccio di lavoro seguito

- Considerazioni: nessuno dei metodi di segmentazione fin qui testati si è dimostrato efficace nella segmentazione accurata di tutti i tipi di immagini. Viene dunque utilizzato il metodo di segmentazione "classificazione dello sfondo".

Pipeline di elaborazione

Pipeline di elaborazione



Metodo di segmentazione

Metodo scelto:

classificazione dello sfondo + active contours

Classificatore: KNN ($k=3$) su deviazione standard con tile di 30×30

300 iterazioni di active contours + Closing morfologico con elemento strutturale a disco di raggio 2

Procedura di segmentazione:

1. Per ogni pixel viene considerato un intorno 30×30
2. Viene calcolata la deviazione standard
3. Il dato viene passato al classificatore

Si ottiene una segmentazione sommaria

1. Viene utilizzato active contours sull'immagine originale partendo dai bordi presenti nell'immagine segmentata.
2. Viene effettuato un closing morfologico

Metodo di
segmentazi
one



raw



output classificazione sfondo output di activeCouturs



raw



output classificazione sfondo output di activeCouturs

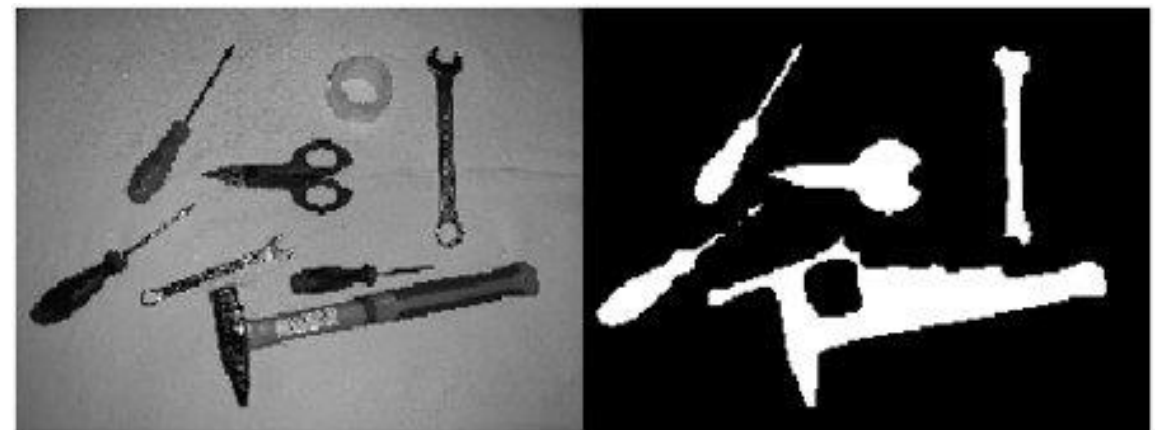
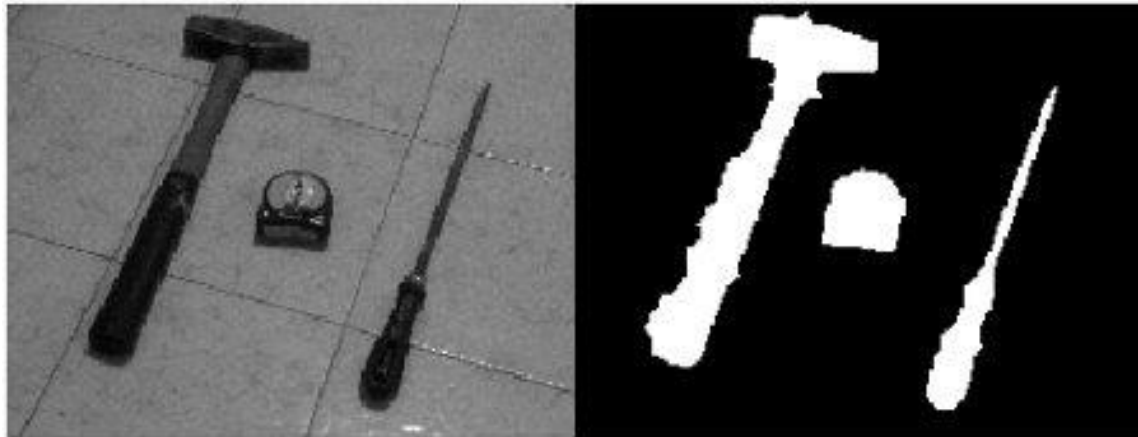


raw



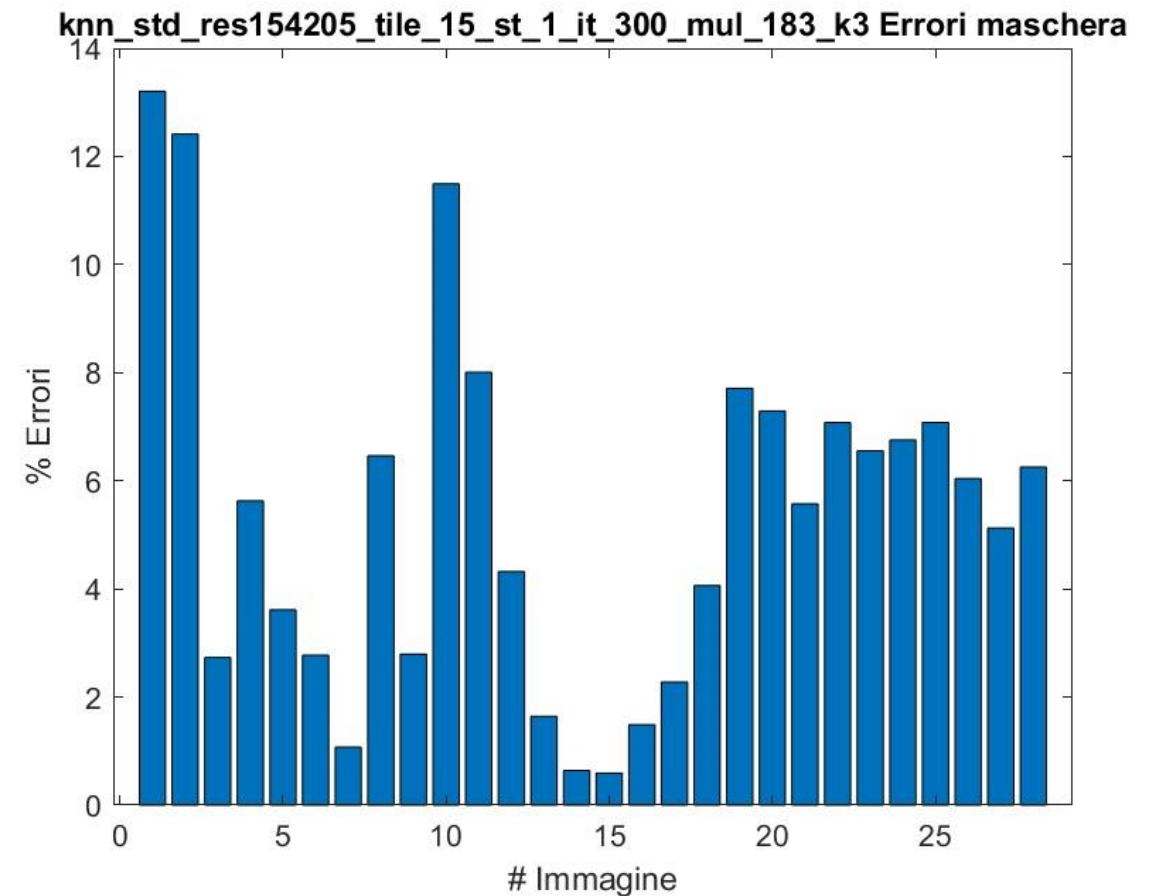
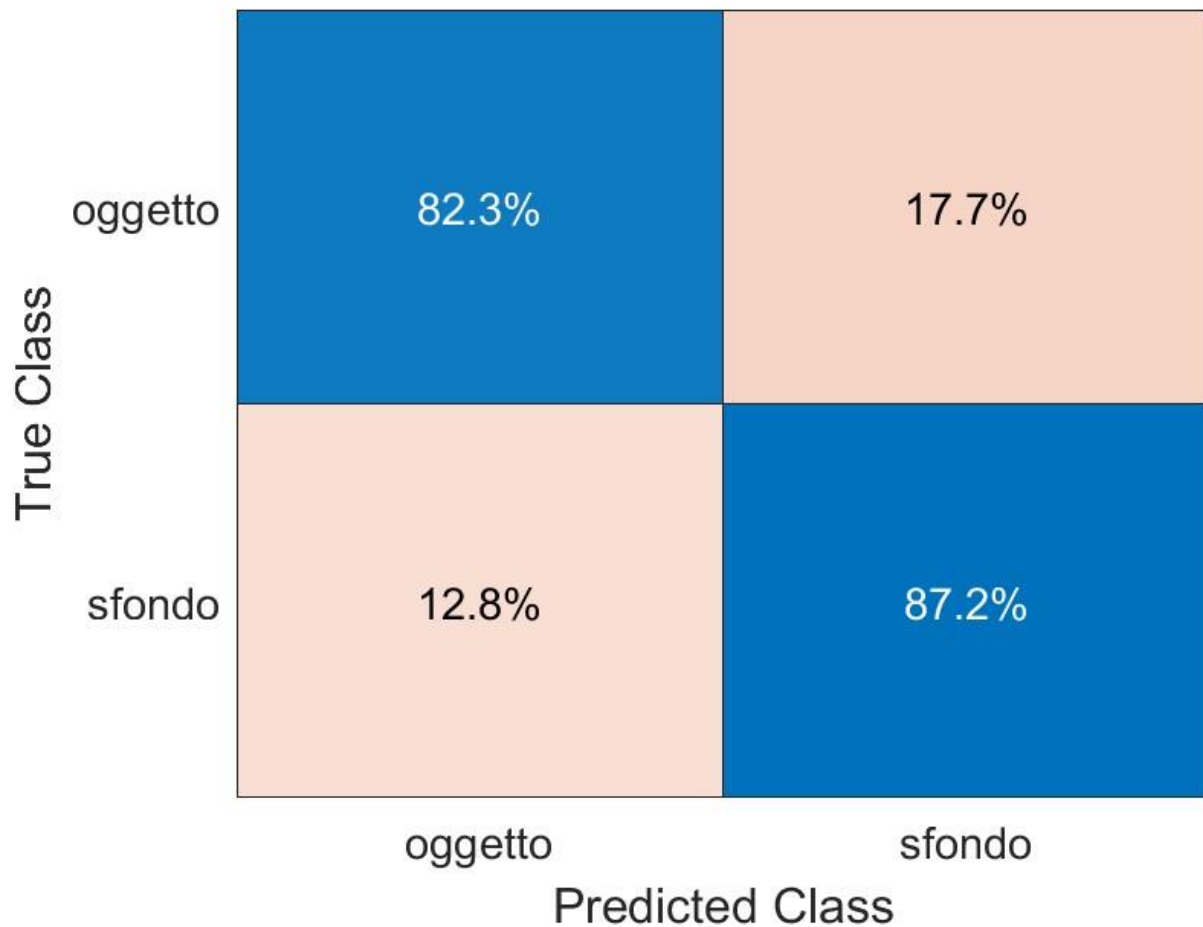
output classificazione sfondo output di activeCouturs

Metodo di segmentazione: risultato finale

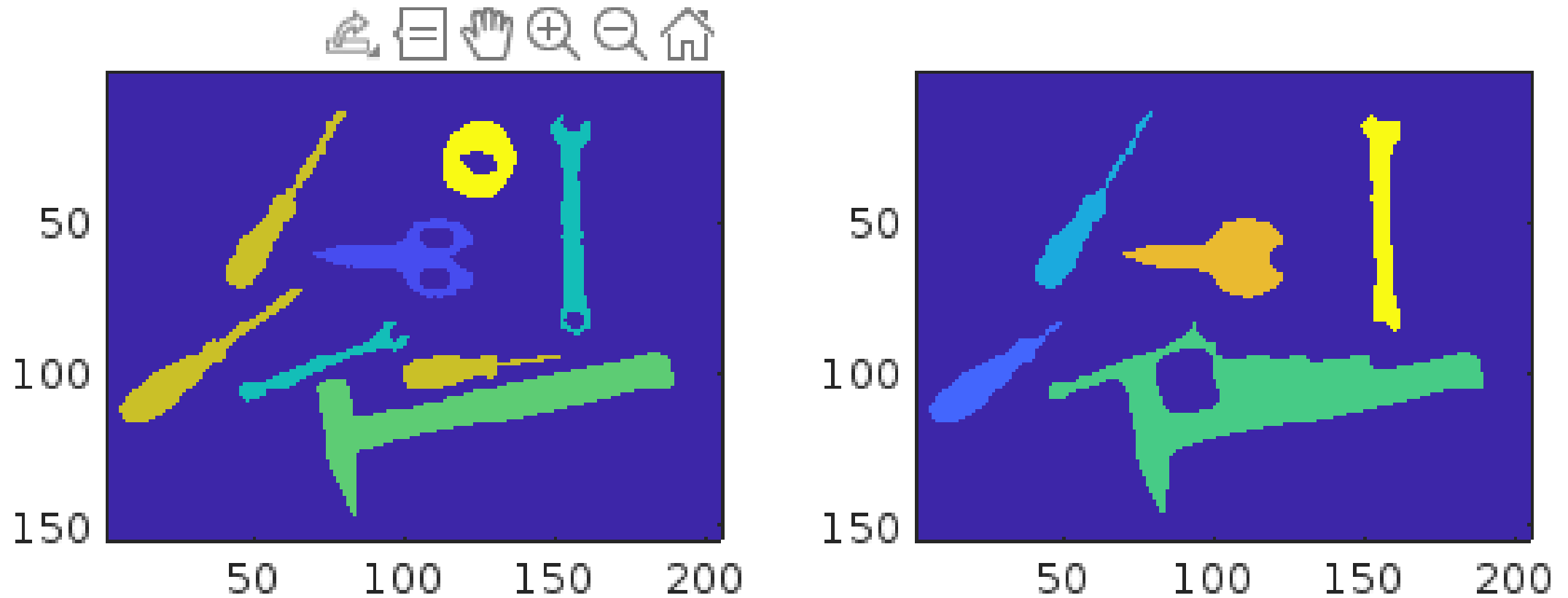


Metodo di segmentazione

Stime numeriche di accuratezza



Labelling Componenti connesse



Risultato Ideale vs. realtà

Features Classificazione

- Rapporto Assi dell'ellissi che include l'oggetto
- Numero di eulero
- Circolarità = $(4 * \text{Area} * \pi) / (\text{Perimeter}^2)$.
- Solidity = numero pixel nel convex hull / num pixel totali oggetto
- Media deviazione standard pixel dell'oggetto
- Momenti di Hu

Nota: le features sono normalizzate ("min-max")

Classifico gli
oggetti:
uso un
classificatore
ad albero

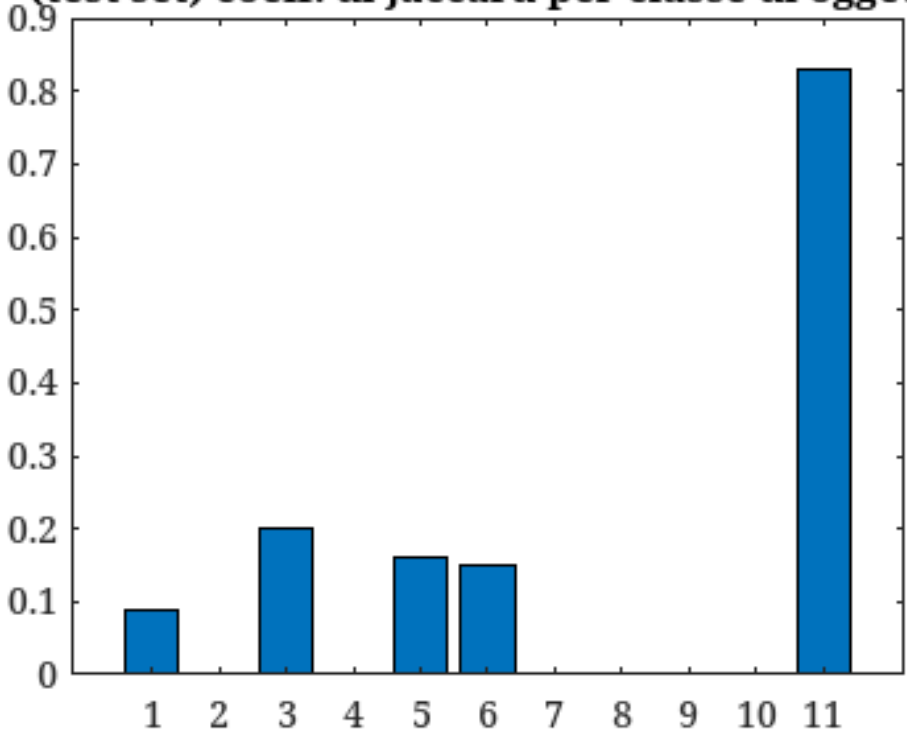


Classificazione oggetti



Classificazione oggetti: risultati su test e train set rispettivamente

(test set) coeff. di jaccard per classe di oggetti



- 1 -> forbice

2 -> metro

3 -> pinza

4 -> chiave

5 -> martello

6 -> cacciavite

7 -> avvitatore
- 8 -> sconosciuto

9 -> pappagallo

10 -> lima

11 -> pennarello.

True Class	avvitatore	66.7%					33.3%				
	cacciavite		66.7%	8.3%		8.3%			8.3%	8.3%	
	chiave				50.0%			50.0%			
	forbice	28.6%		14.3%	28.6%		14.3%	14.3%			
	lima					50.0%				50.0%	
	martello				27.3%		72.7%				
	metro	9.1%			18.2%	9.1%		18.2%	27.3%		18.2%
	pappagallo				42.9%		14.3%		28.6%	14.3%	
	pennarello		33.3%							66.7%	
	pinza		14.3%					14.3%			71.4%

avvitatore

cacciavite

chiave

forbice

lima

martello

metro

pappagallo

pennarello

pinza

accuracy: 0.4923

Predicted Class

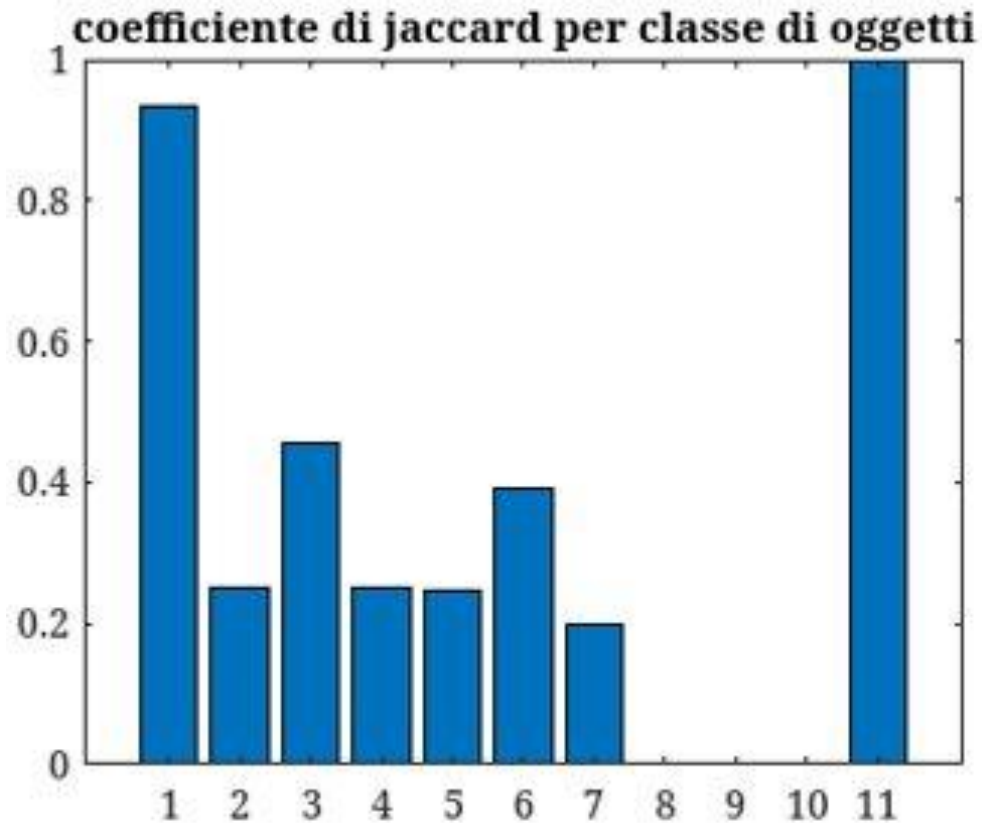
Alcune classi hanno prestazioni molto scarse, probabilmente dovute alla minor presenza di campioni di tali classi nel set di test, causando così errori nella classificazione.

Conclusioni

Conclusioni

- Numero elevato di classificazioni sbagliate
- Features per la classificazione non abbastanza discriminanti
- Dataset poco numeroso per alcune classi
- Segmentazione visivamente accettabile (comunque migliorabile), ma limitante per quanto riguarda la classificazione di oggetti.

Coeff. di jaccard sul test set, come segmentazione uso la ground truth:



Ricorda che il classificatore non è allenato sul test set e quindi non devo aspettarmi risultati perfetti.

Contributo

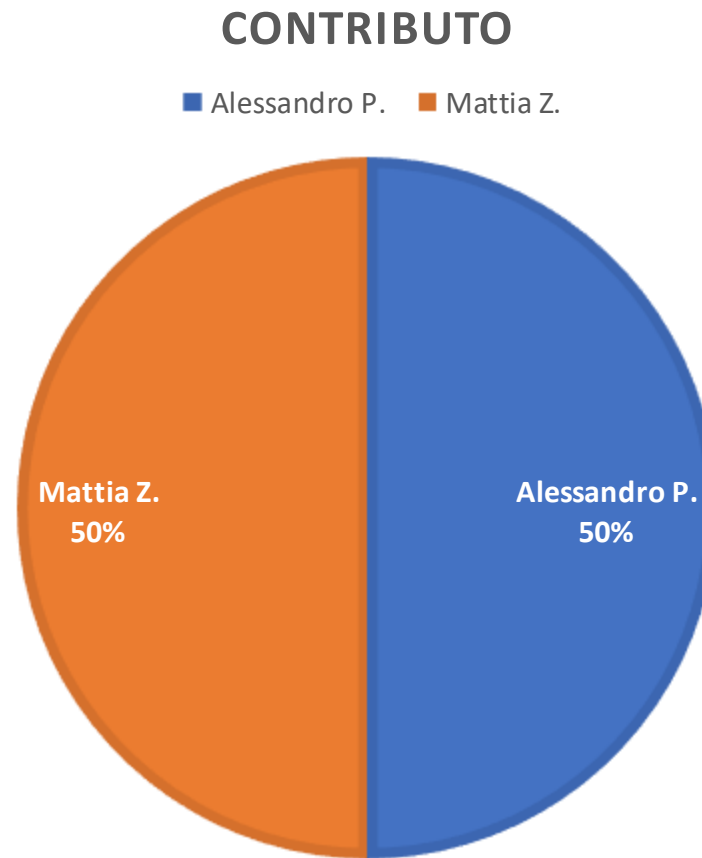
Mattia Zirpoli

- Funzione calcolo momenti di Hu
- Creazione maggior parte delle Ground Truth Singole
- Segmentazione manuale GT multiple
- Esperimenti segmentazione e classificazione (iniziale) (segmentazione morfologica, thresholding su intervallo)
- Funzione visualizzazione percentuale errori (bargraph)

Alessandro Preziosa

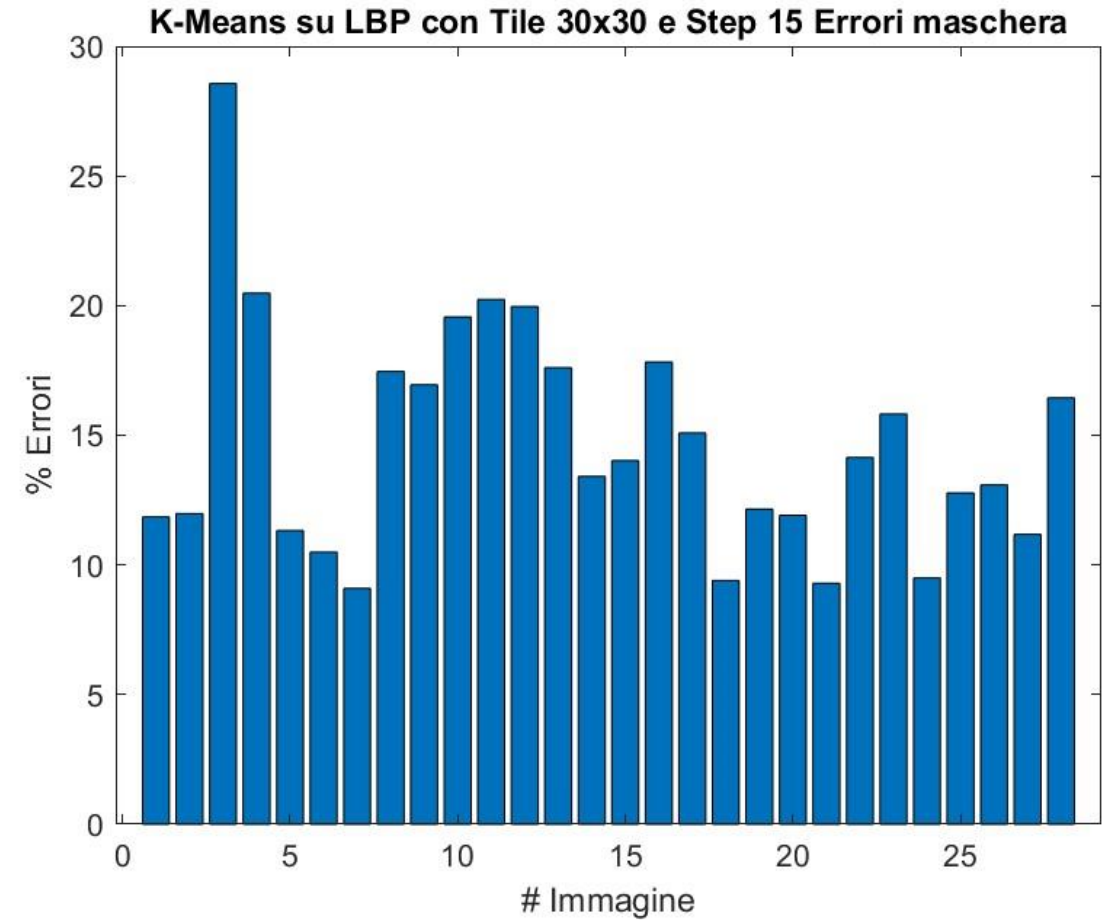
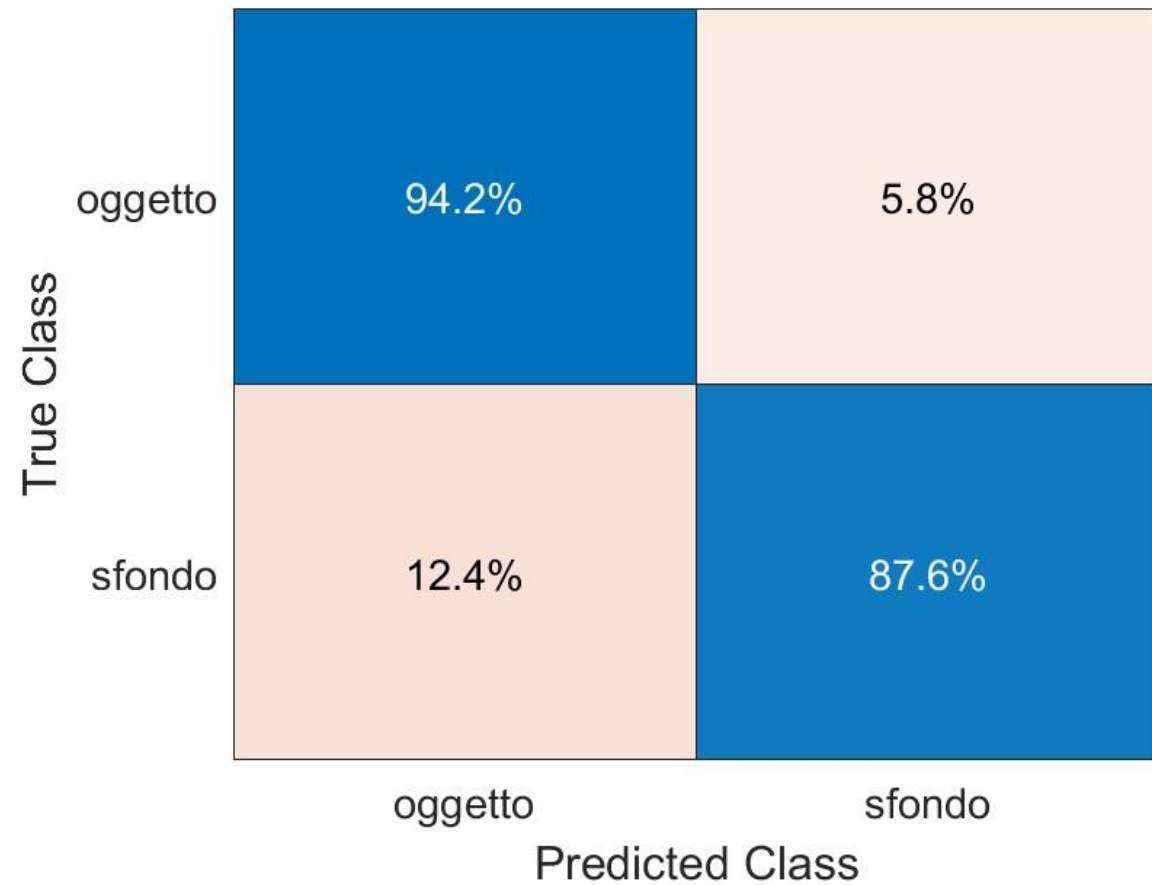
- Funzioni valutazione segmentazione e classificazione
- Trainer classificatori sfondo e oggetti
- Funzione Estrazione features
- Calcolo alcuni descrittori
- Funzione d'appoggio per lettura immagini
- Esperimenti segmentazione(kmeans, segm. Via classificaz, sauvola, thresholding globale)

Contributo



Diapositive di Supporto

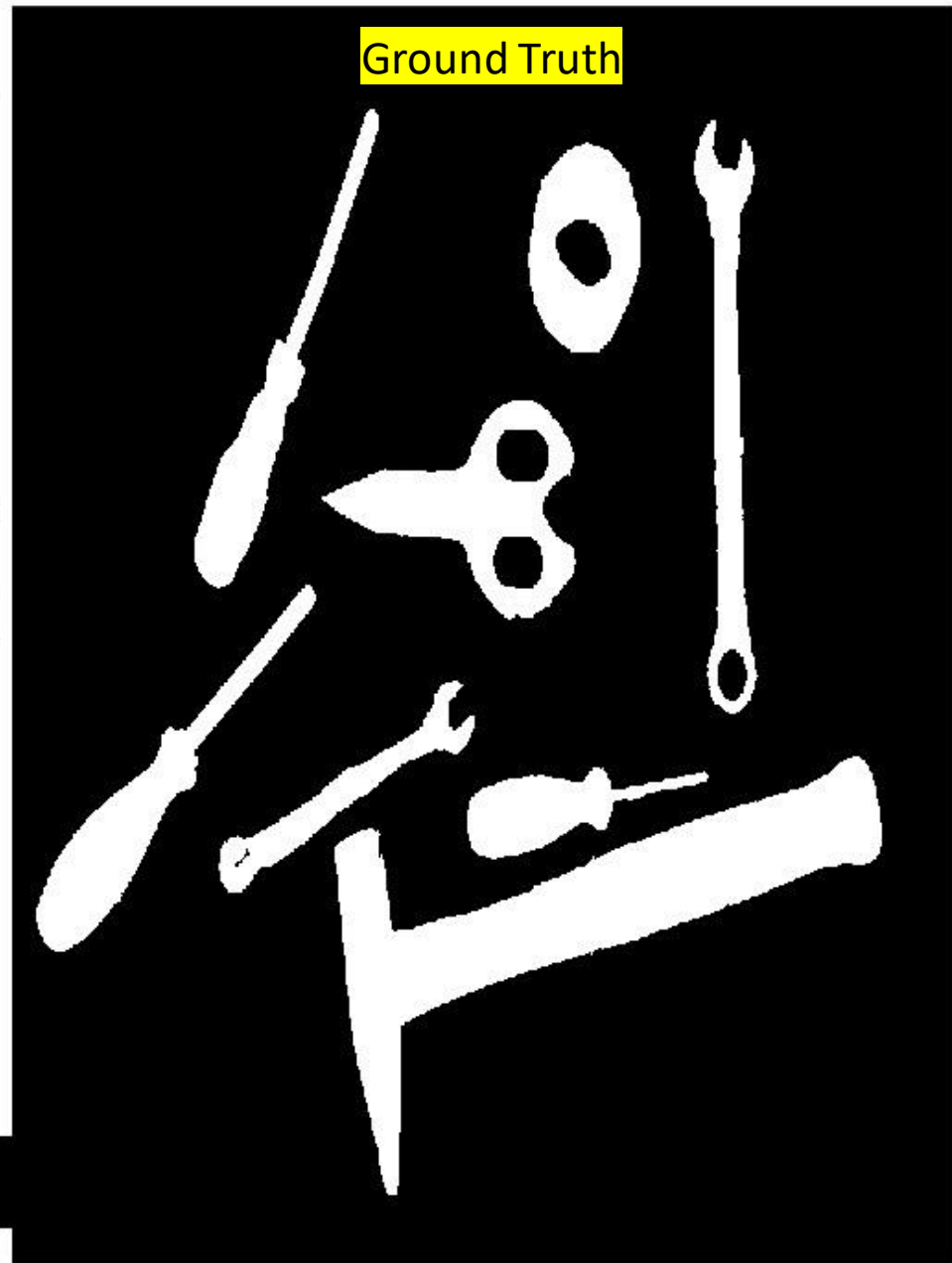
Segmentazione K-means su LBP (tile 30x30, step 15)



K-means



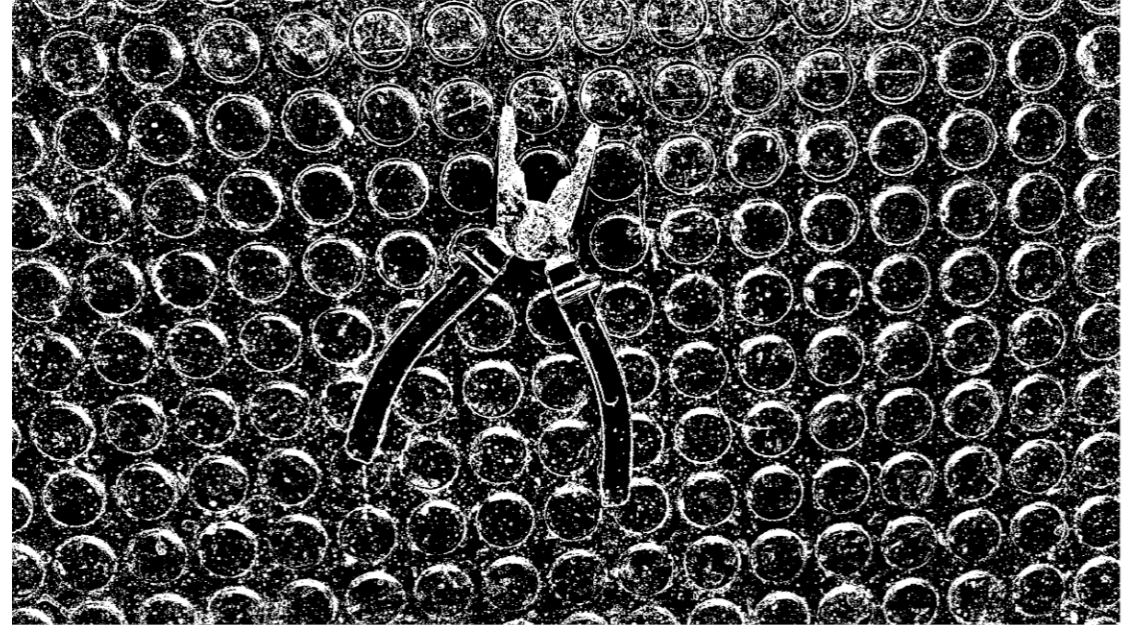
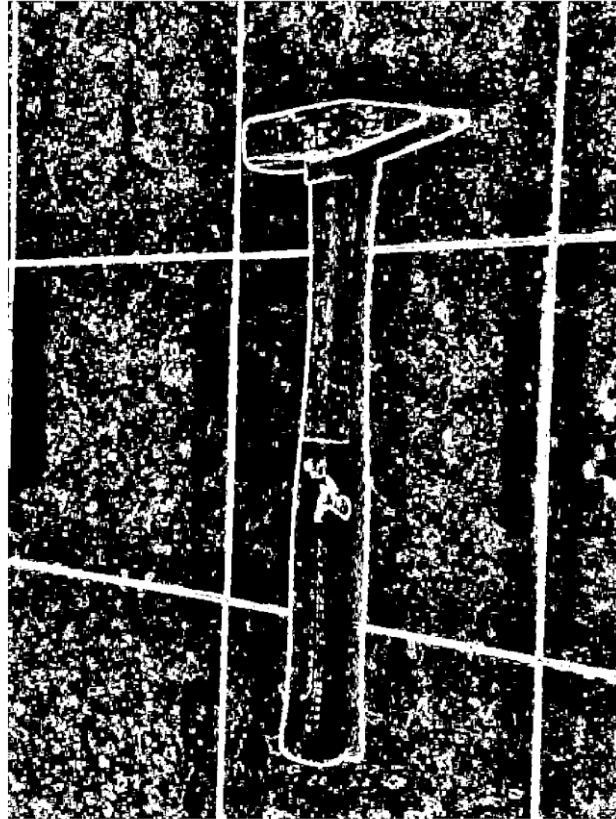
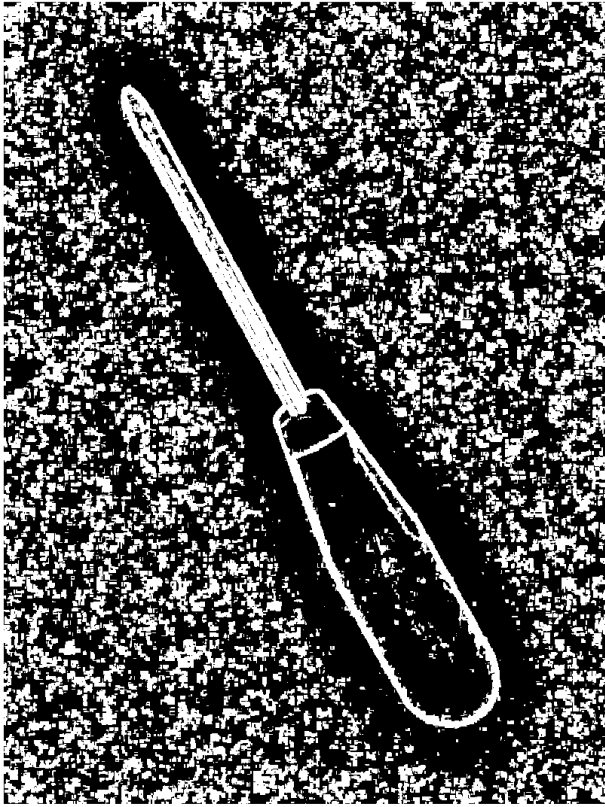
Ground Truth



Segmentazione via Morfologia

- **Ipotesi:** Separare oggetto e sfondo velocemente
- **Metodo:** Binarizzazione adattiva dello smoothing morfologico, con apertura della binarizzazione
- **Conclusioni:** Risultati interessanti, ma inutilizzabili per un'effettiva segmentazione

Risultati:



Segmentazione con gradiente

Originale



Gradiente

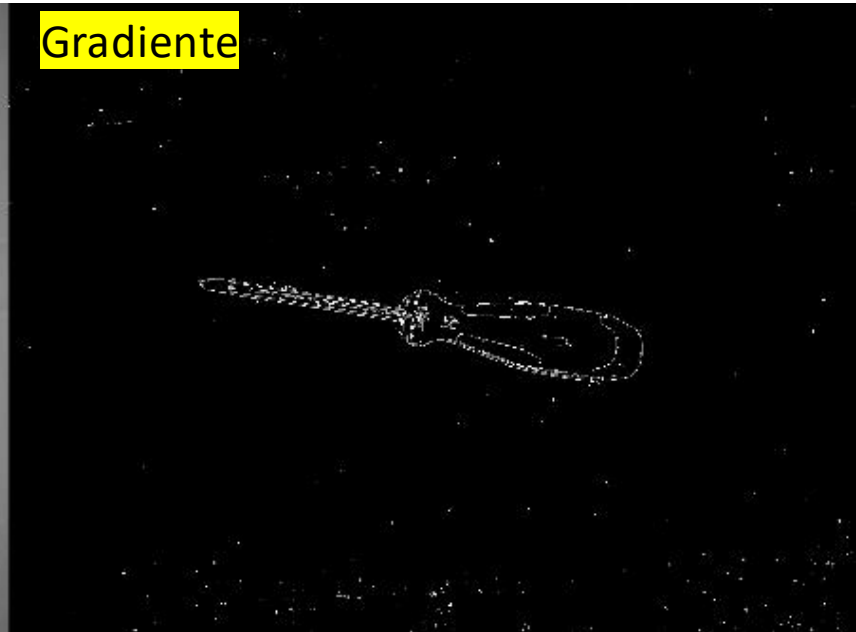
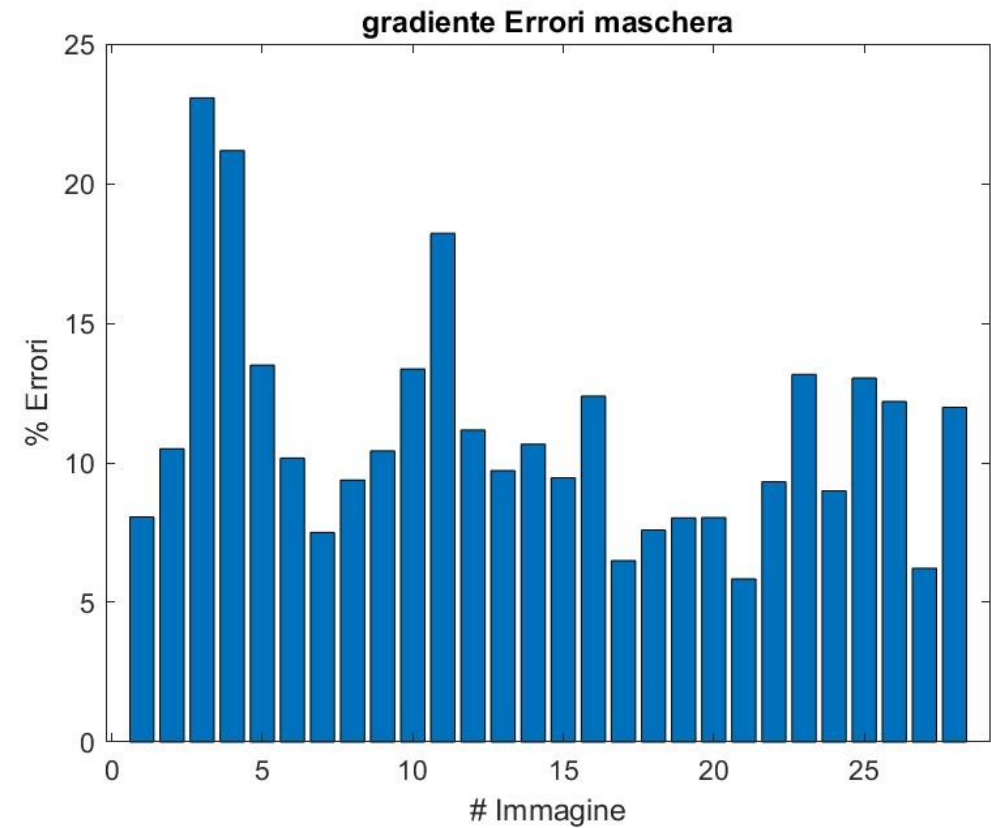
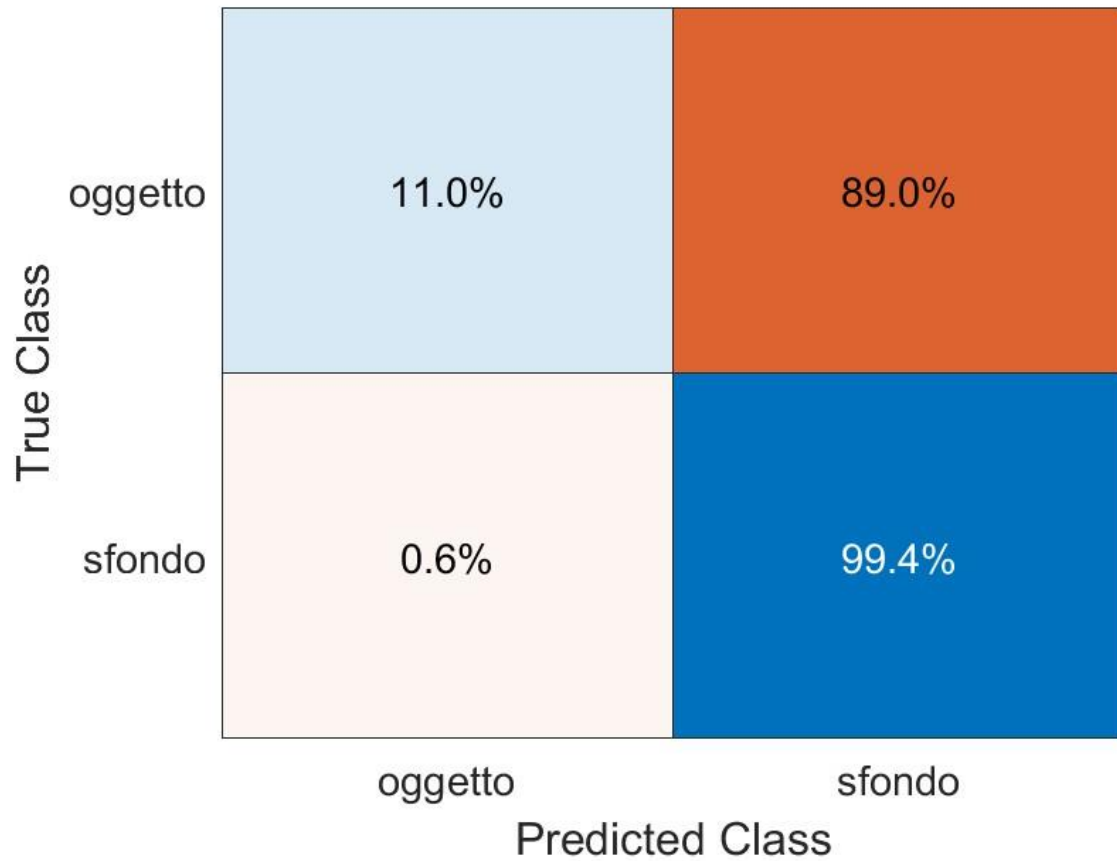


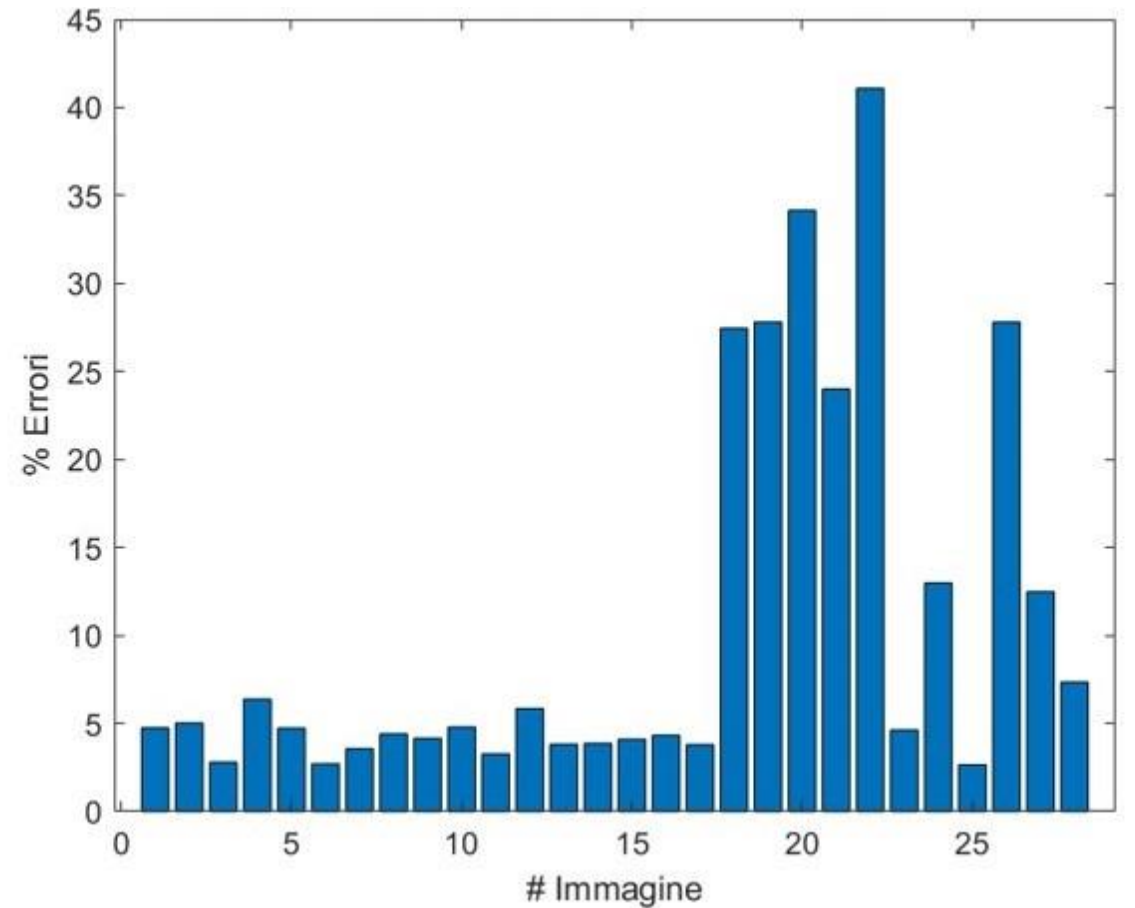
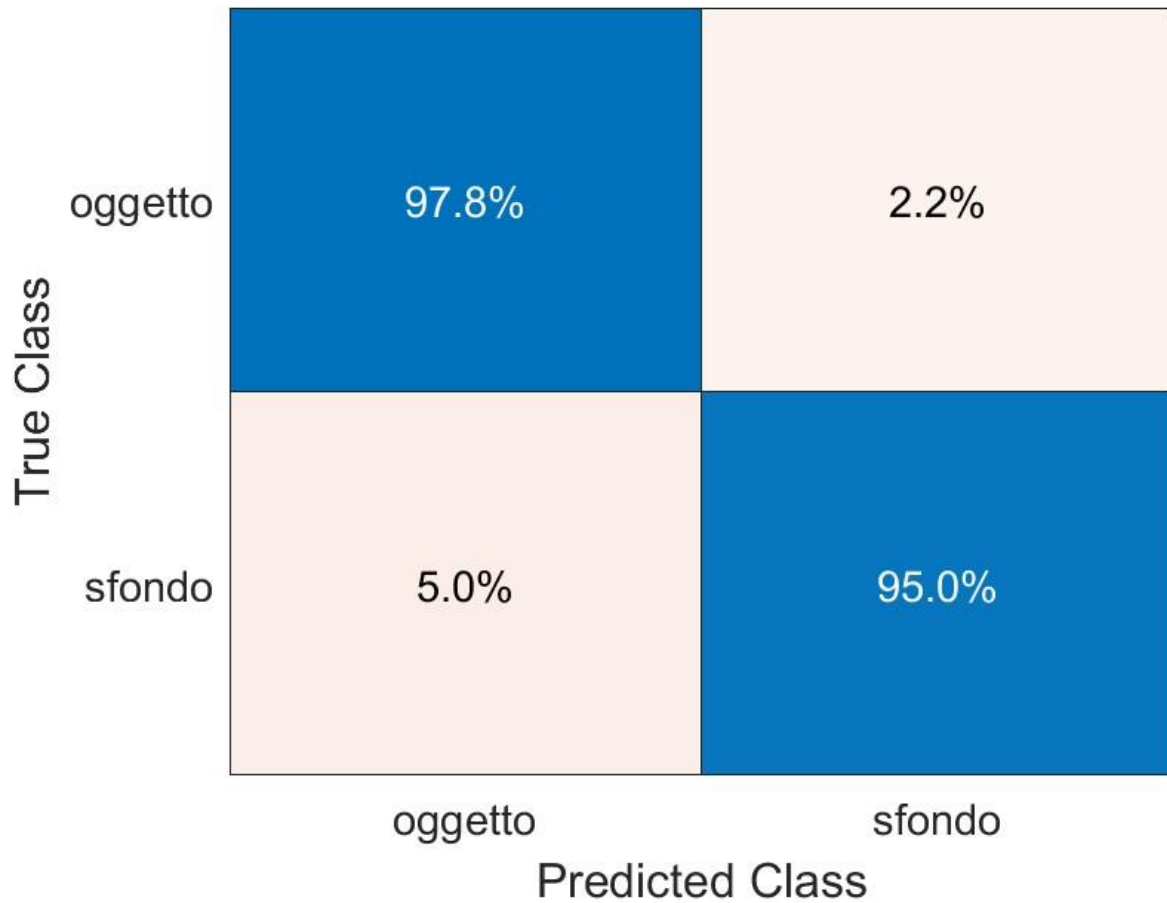
Immagine moltiplicata per Gradiente



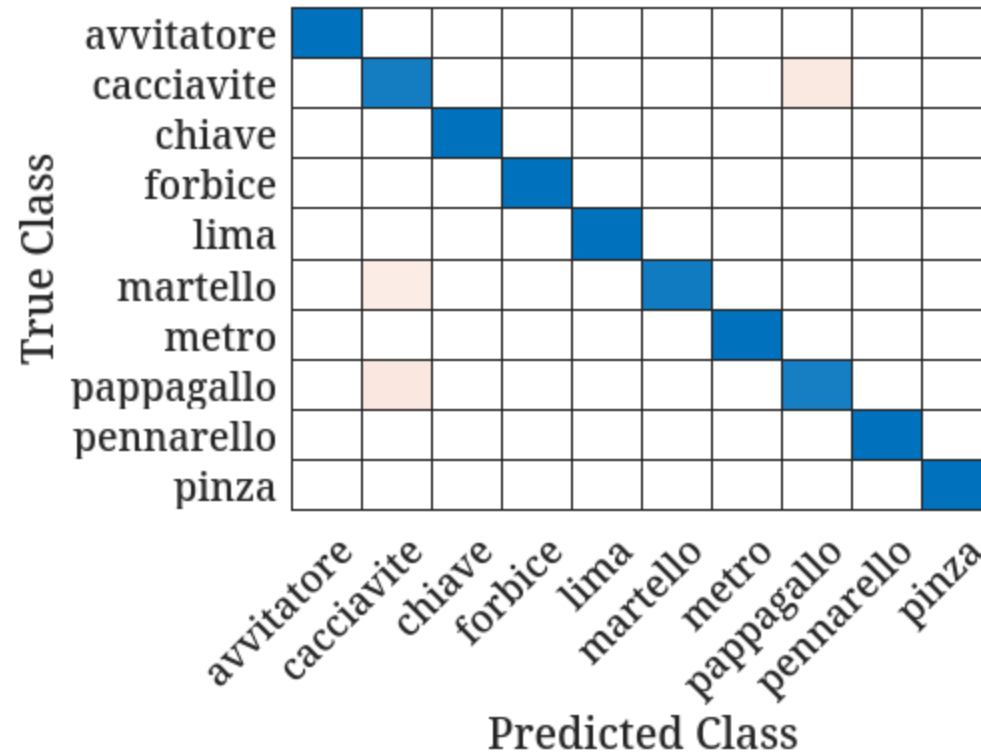
Segmentazione con gradiente



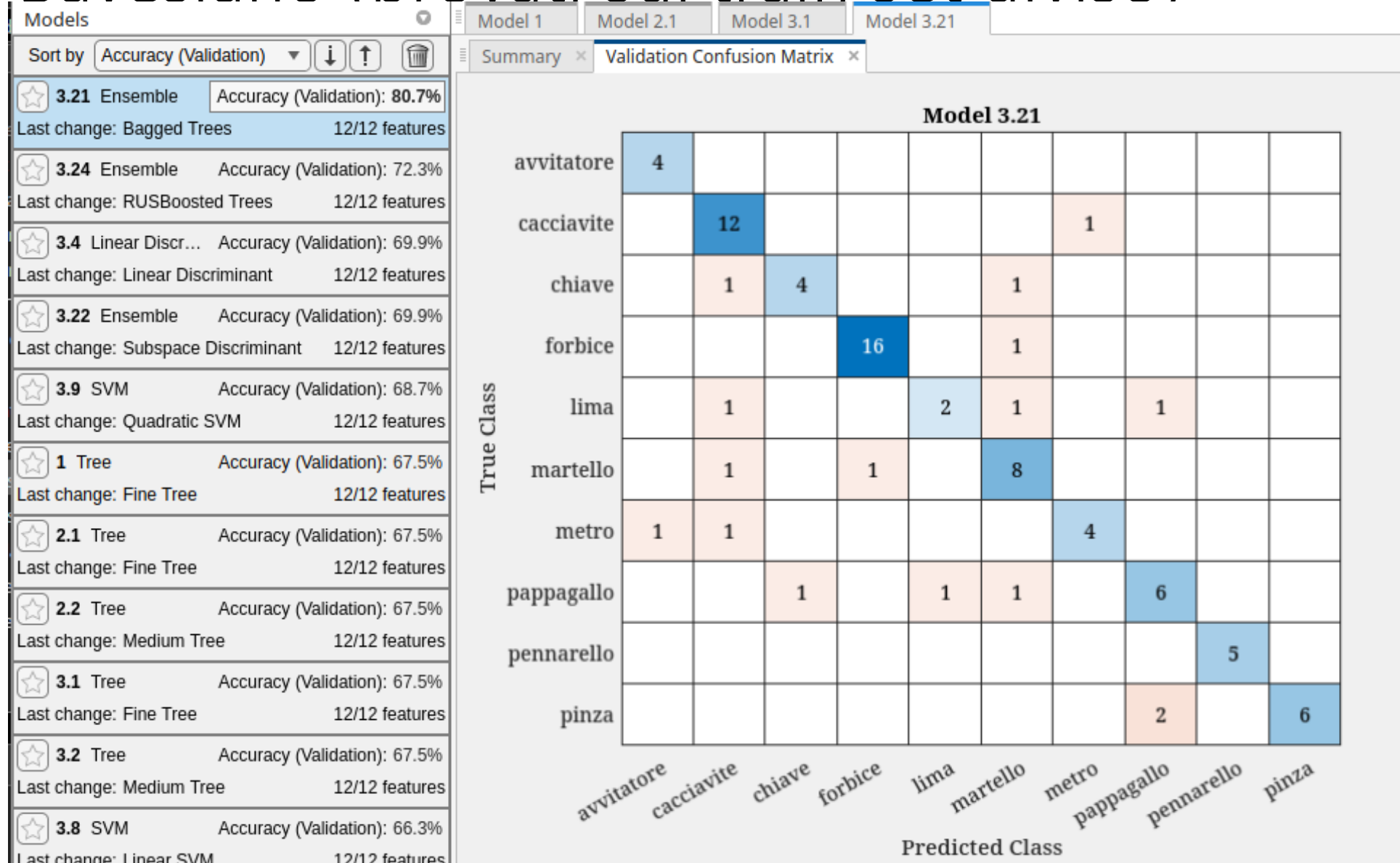
Segmentazione Sauvola



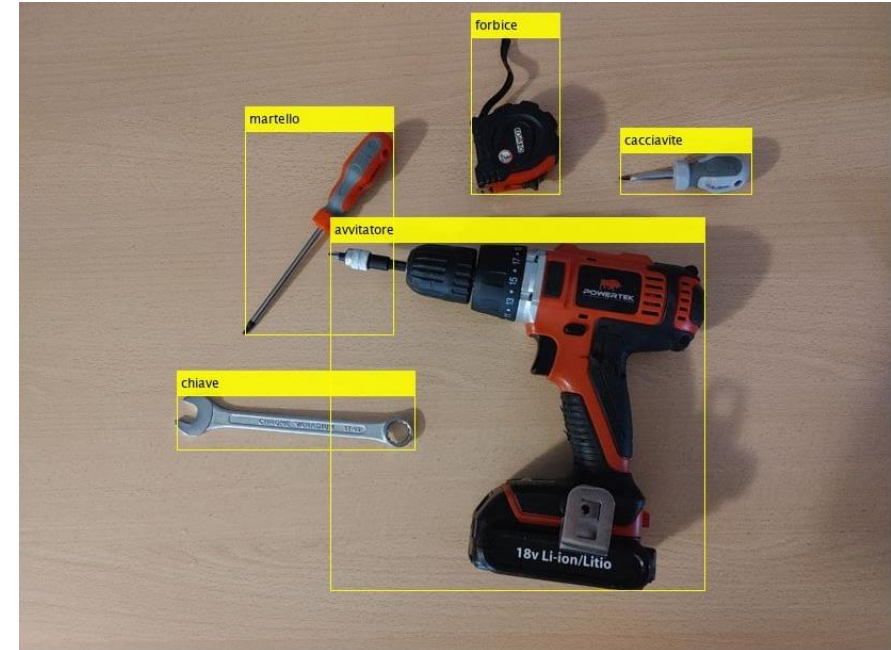
Classificazione oggetti (train set) con la gt al posto della segmentazione



Altri Classificatori: Ensemble, SVM, Tree, Neural Network, KNN, Bavesiano (provati sul train set diviso)



Prove con classificatori di oggetti: Classification Tree



Altri
classificatori di
oggetti:
KNN (k=10)



KNN

- Le immagini sono tutte riscalate a 154x205px
- Il training usa come feature la deviazione standard
- La deviazione standard è calcolata su tile 15x15 usando uno step di 1px
- $K = 3$

Funzionamento Algoritmo KNN:

Un oggetto da classificare è un punto nello spazio n-dimensionale, con ogni dimensione che corrisponde ad una feature. Uso il train Set per piazzare punti nello spazio ed associare loro un etichetta(classe).

Per classificare dati(punti) mai visti (senza l'etichetta) ... vado a guardare i suoi k punti più vicini:

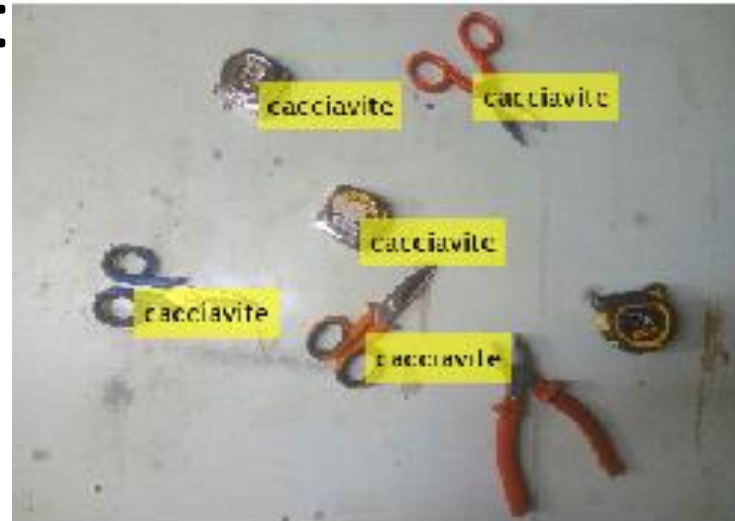
Guardo la loro etichetta e quella più popolare tra i k vicini sarà quella che darò al punto.

Solo i dati del train set "votano" per le label degli altri.

Altri classificatori: optimized cTree



Altri classificatori:
automatico

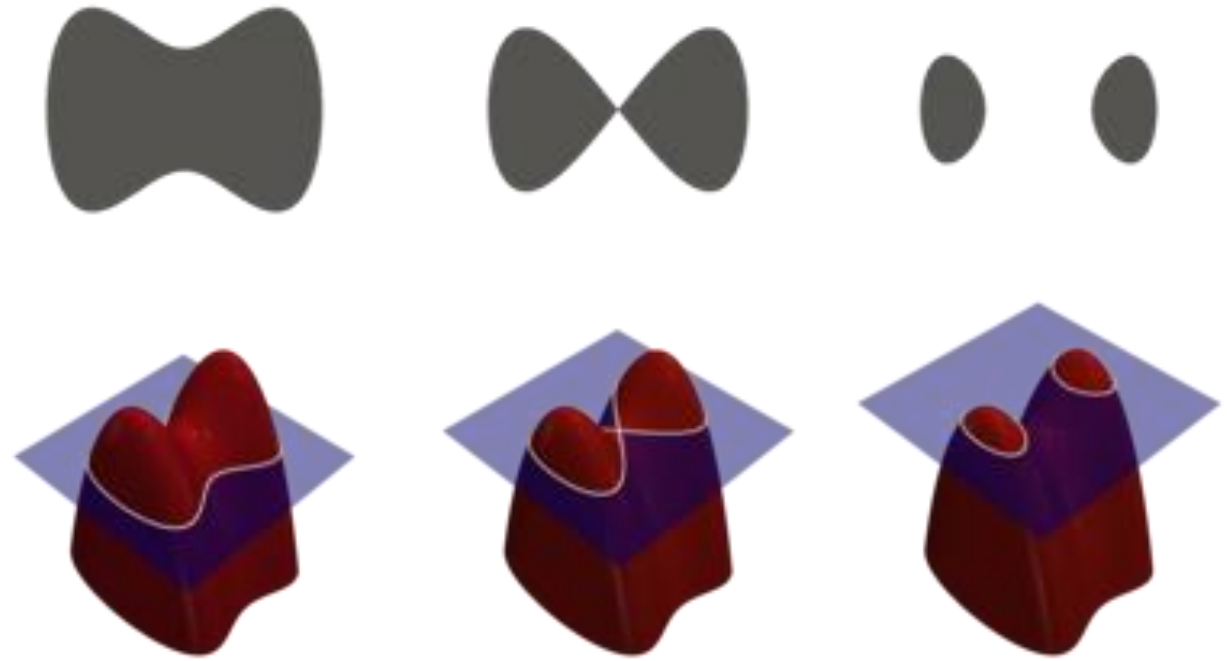


Funzionamento Active Contours

- Utilizza l'algoritmo di Chan-Vese
- Partiziona l'immagine in due regioni – background e foreground
- Definisce una «level-set function»
- La level-set function viene evoluta tramite l'equazione di Eulero-Lagrange
- L'obiettivo è minimizzare una funzione «energia» composta da due termini
 - Energia interna – «Smoothness»
 - Energia esterna – «Differenza tra regioni»

Active Contours – Level Set Function

- Descrive il contorno tra le regioni di background e foreground
- Minimizza la funzione di energia in base all'equazione di Eulero-Lagrange
- Si ferma in due casi
 - Massimo di iterazioni raggiunto
 - Funzione energia minimizzata



Equazione di Eulero-Lagrange

- Descrive come dovrebbe cambiare la Level-Set Function ad ogni iterazione per minimizzare la funzione energia
- È un'equazione differenziale alle derivate parziali
- Garantisce che la soluzione converga al minimo locale della funzione «energia»

