

Leaf-ID

Un Progetto di Teodori Alessandro e
Rossi Andrea



Indice

01

Caratteristiche Progetto

- Obiettivo del Progetto
- Caratteristiche del Dataset
- Problemi da Affrontare

02

Pipeline del Progetto

- Acquisizione delle Immagini
- Segmentazione
- Calcolo delle Features
- Riconoscimento delle Immagini
- Classificazione delle Foglie

03

Conclusioni

- Risultati
- Idee per miglioramenti

Obiettivo del Progetto



Riconoscere, Localizzare e Classificare

Lo scopo del progetto è realizzare un algoritmo di Leaf classification con almeno 10 classi.

Gli oggetti possono essere catalogati in una delle 10 classi o come unknown

Problemi da Affrontare



Problemi

- Oggetti estranei nell'immagine.
- Riflesso nelle foglie
- Tinta chiara delle foglie
- Foglie Ruotate
- Foglie con Forma e texture simile

01-Acquisizione delle Immagini



Caratteristiche del Dataset



Le Immagini presenti nel dataset e usate nel testing presentano le seguenti caratteristiche.



Classi

Il dataset è composto da 10 classi +1 Classe Sconosciuta per gli oggetti che non rientrano nelle 10 classi



Illuminazione

L'illuminazione nelle immagini di training e test è costante per direzione e variabile per intensità.



Sfondo

Lo sfondo nelle immagini di training e testing è bianco con una texture uniforme.



Free corner

Per lo scopo del Progetto si presuppone che il gruppo di pixel 10x10 più in alto a sinistra sia sempre libero.

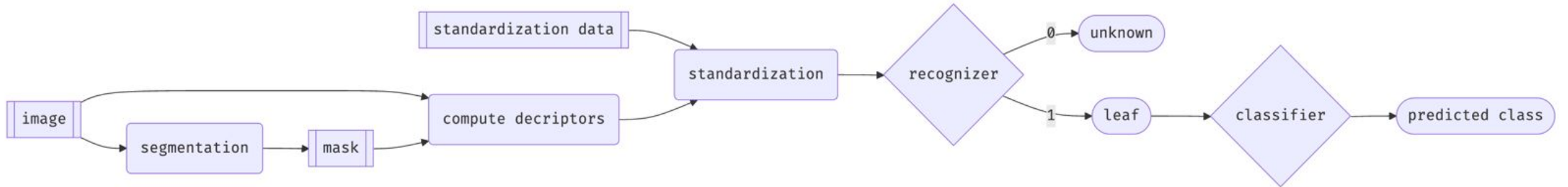


Sovrapposizione

Si presuppone che nessun oggetto sia sovrapposto a un altro e che nessun oggetto venga tagliato dal bordo.



Project Pipeline



02-Segmentazione delle Immagini



Region-Growing con LAB



Come abbiamo implementato la segmentazione degli oggetti?

Il programma che abbiamo implementato converte una immagine fornita in spazio colore LAB. In congiunzione con un algoritmo di region-growing possiamo discriminare i vari pixel tra sfondo e oggetti.

1. Come mai proprio LAB?

È lo spazio colore più adatto per confrontare le differenze percepite visivamente.

2. Come discrimina l'algoritmo?

Il metodo region-growing che abbiamo implementato calcola la distanza nel colore LAB rispetto al seme con un peso di 0.5 su L. Se la distanza è inferiore alla soglia (settata a 21), il pixel viene aggiunto al background.

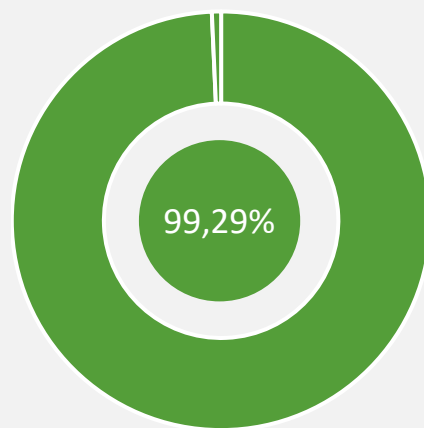
3. Perché dare meno peso a L?

Testando abbiamo notato che gli oggetti con tinte più scure venivano incluse nello sfondo. Per evitare ciò abbiamo voluto ridurre il peso di L

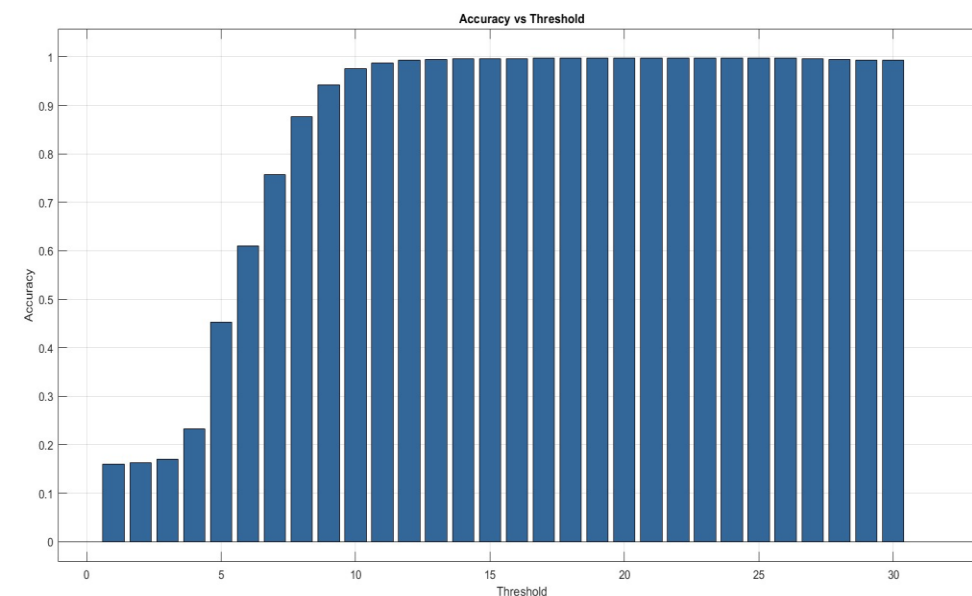
Perché una Soglia di 21?

Il valore di soglia non è stato scelto a caso.

Inizialmente testando abbiamo optato per una soglia intorno ai 18 perché dava i risultati miglior. Ma, dopo aver applicato un algoritmo che paragonava l'accuratezza del segmentatore su 30 soglie diverse abbiamo appurato che la soglia migliore fosse 21.



La percentuale di accuratezza
del segmentatore con soglia
21



Segmentatori Differenti

Prima di scegliere region-growing sono stati considerati altri metodi, che però non soddisfacevano alcune delle nostre esigenze

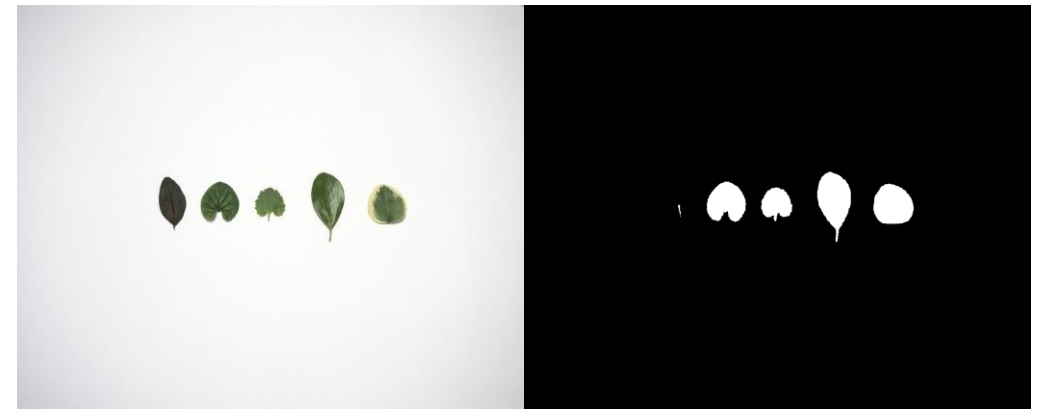
1. K-means

Abbiamo scartato il clustering per problemi nel riconoscere i pixel di oggetti come parte dello sfondo

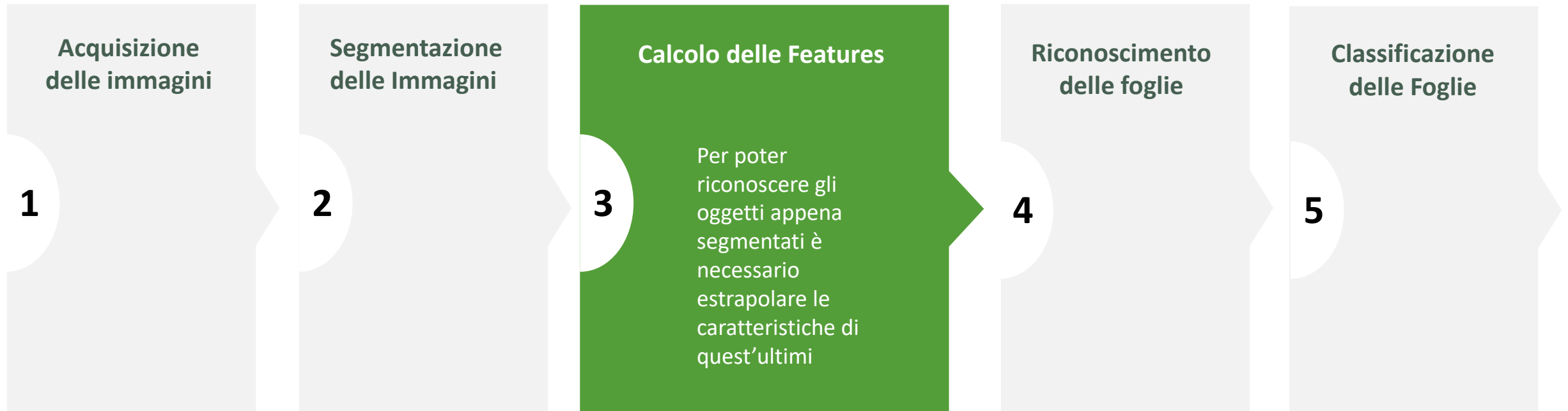


2. Region-Growing con i canali a e b

In precedenza dello spazio colore Lab abbiamo escluso la L ma dopo aver notato che alcuni oggetti molto scuri o alcune foglie molto chiare venissero incluse nello sfondo, abbiamo ritenuto necessario includere anche lo spazio L



03-Calcolo delle Features



Calcolo delle Features

Forma/Edge

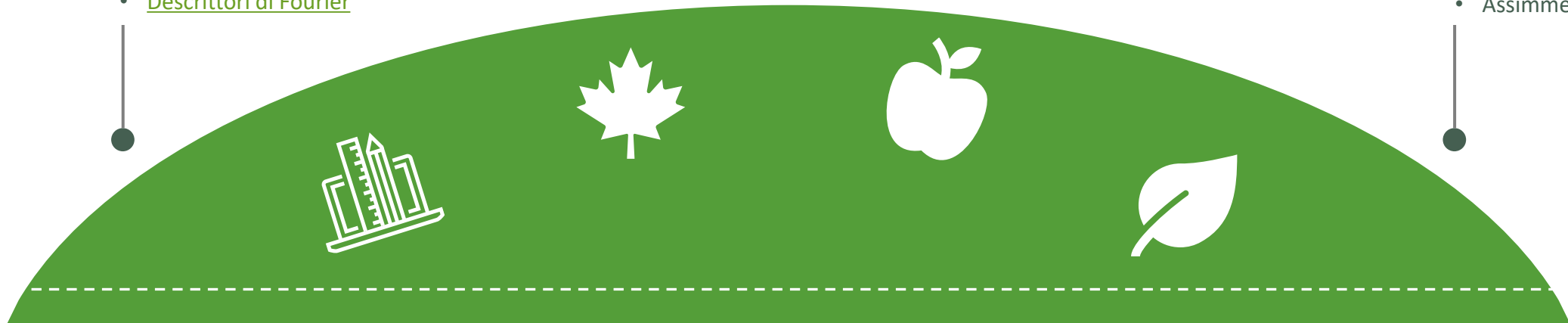
- Signature Polare
- Feature Statistiche
- Momenti di Hu
- [Descrittori di Fourier](#)

Texture

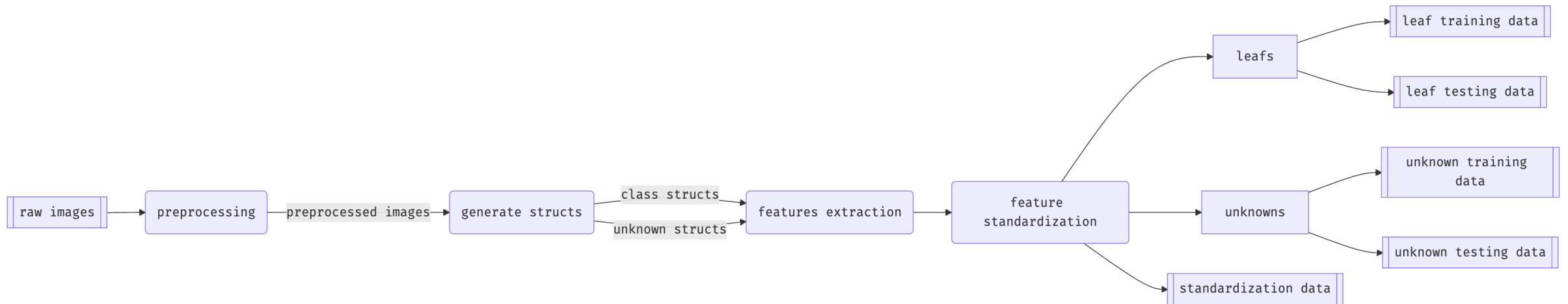
- [RILBP](#)
- Edge Histogram Statistic Features
- [Momenti di Zernike](#)

Colore

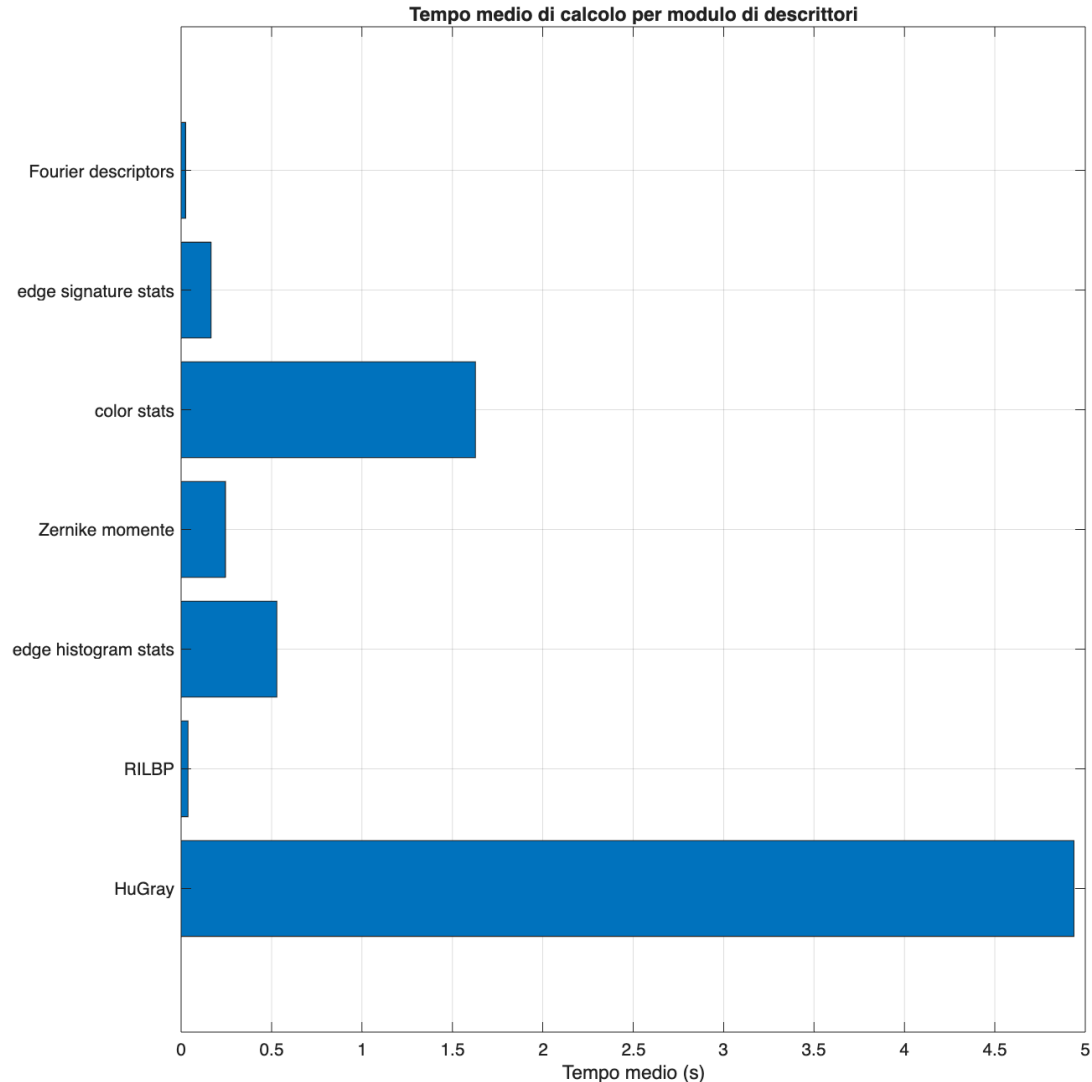
- Media
- STD
- Kurtosi
- Assimmetria Canali



Pipeline di estrazione delle features



Tempi di calcolo per l'estrazione dei descrittori



I seguenti tempi di calcolo sono stati ricavati da un'analisi basata sui risultati del profiler di MATLAB sull'estrazione dei descrittori sui set di training e testing.

È stata inoltre sfruttata la Parallel Computing Toolbox, per distribuire al meglio il carico del calcolo dei descrittori sugli insiemi di train e test.

Selezione Automatica delle Features

È stata effettuata una selezione automatica delle feature. Lo scopo principale di tale scelta è quello di ridurre la dimensionalità del vettore delle features, in modo da alleggerire il carico computazionale nei confronti di riconoscitore e classificatore. La selezione automatica sfrutta un loop RFE e un ranking delle features guidato da MRMR o oobPermutedImportance, in base al modello di target

Ranking delle features basato su MRMR o oobPermutedImportance

Rimozione delle coppie ad alta correlazione

RFE loop con modello a scelta (default ensemble)

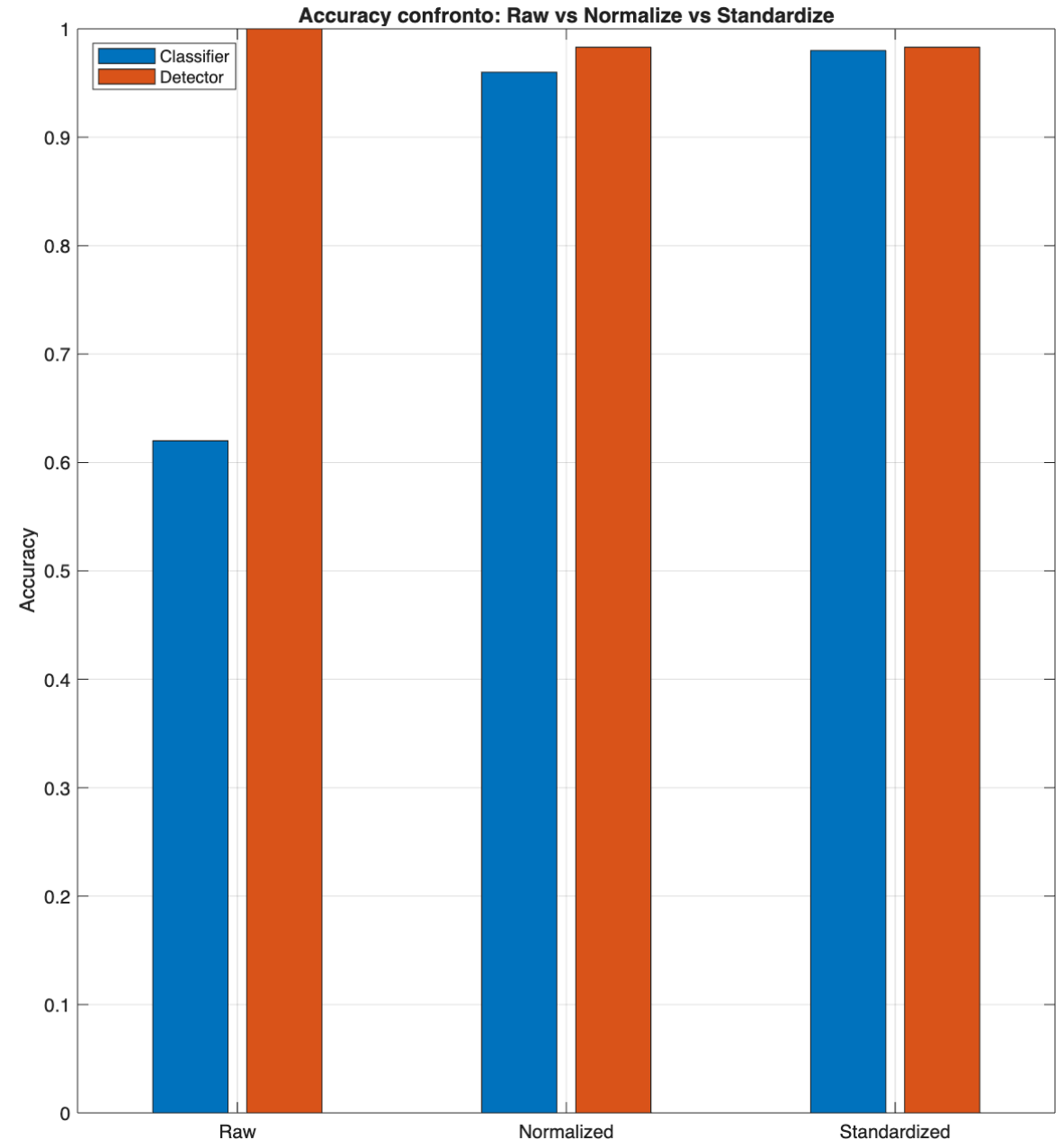


Scaling dei Dati

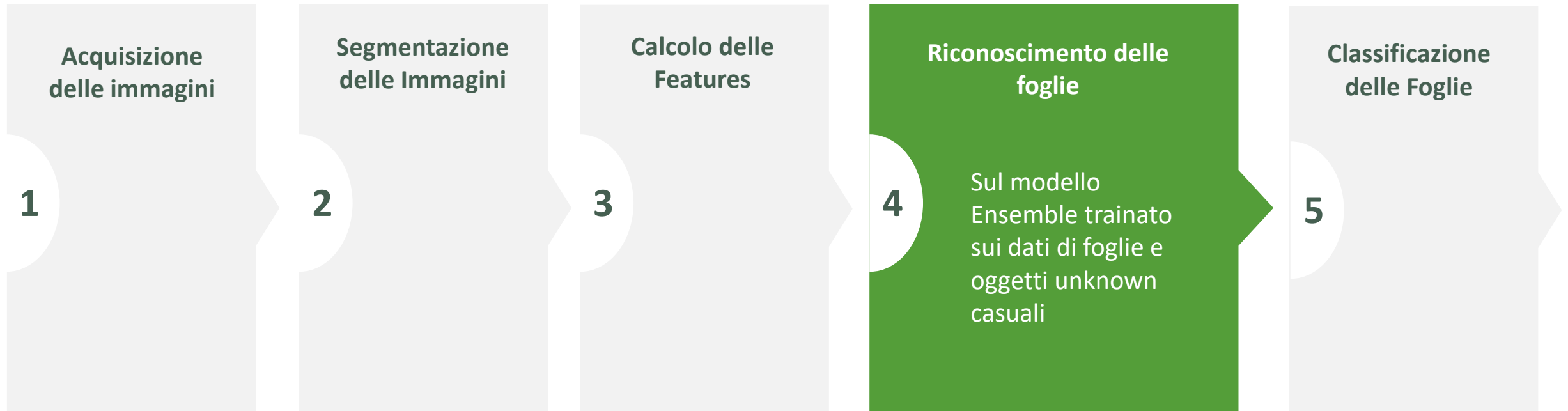
Grafico:

- Dati Raw
- Dati Normalizzati
- Dati Standardizzati

Utilizziamo una standardizzazione per migliorare la performance di KNN nonostante i dati raw siano migliori per il riconoscitore



04-Riconoscimento delle foglie



Come mai un riconoscitore separato dal classificatore?

Come vedremo nella sezione [successiva](#), si è scelto di utilizzare un classificatore di tipo KNN, il quale offre numerosi vantaggi, tuttavia risulta **insensibile agli outliers**.

Ciò significa che, per quanto un oggetto possa presentare features differenti da quelle estratte dalle foglie, non verrà mai riconosciuto come sconosciuto, ma verrà assegnato alla classe con elementi dalle caratteristiche più simili.

Per questo motivo è stato necessario implementare un riconoscitore per permettere al sistema di distinguere foglie da oggetti **unknown**.

Questa decisione consente, inoltre, di mantenere alta la **modularità** del progetto.

Ensemble

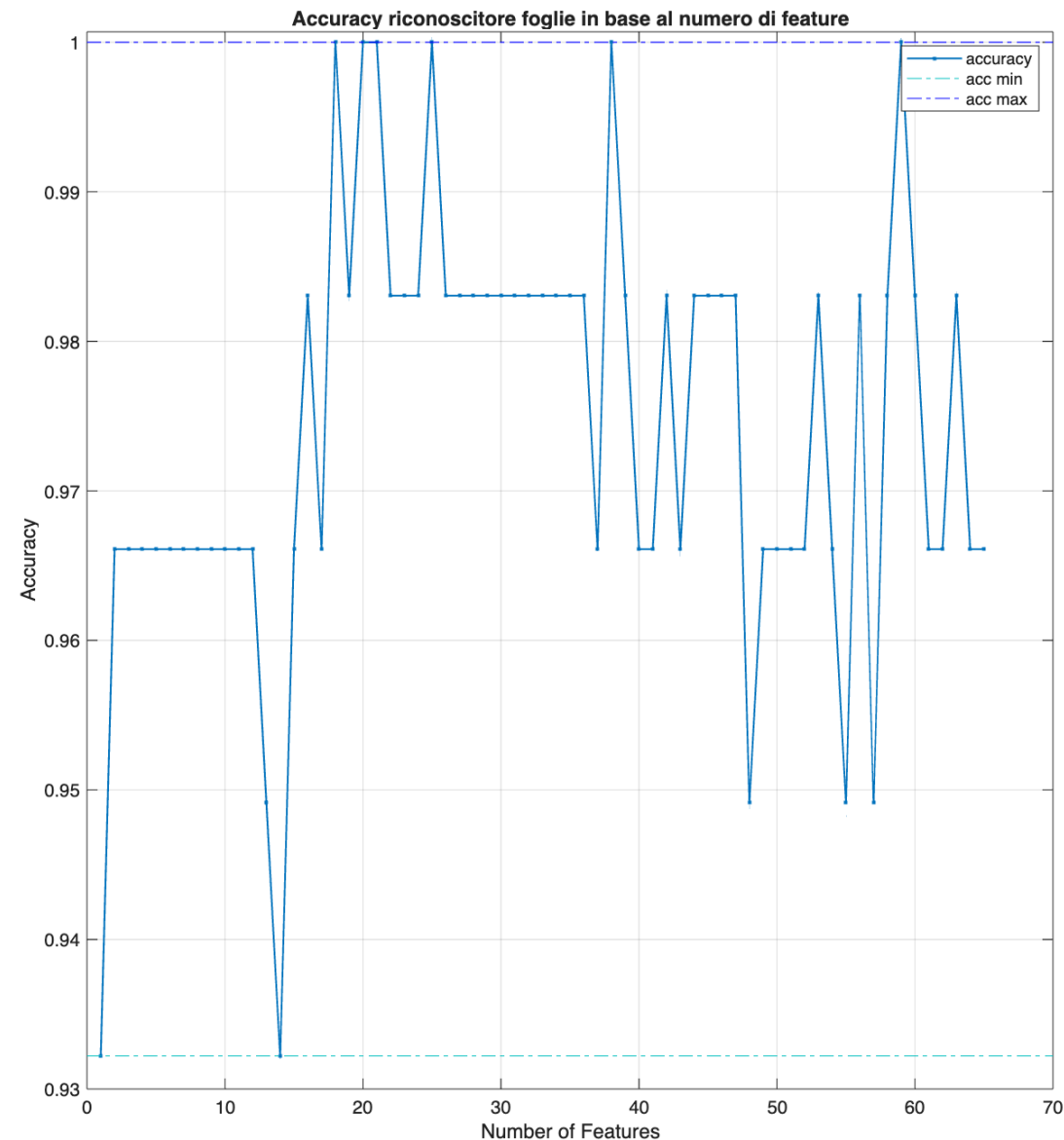
Un riconoscitore ensemble combina le decisioni di più classificatori (random forest nel caso del nostro riconoscitore di foglie) per migliore robustezza e accuratezza

Ogni classificatore fa una predizione e la decisione è ottenuta tramite voto o media delle predizioni (in base al metodo)

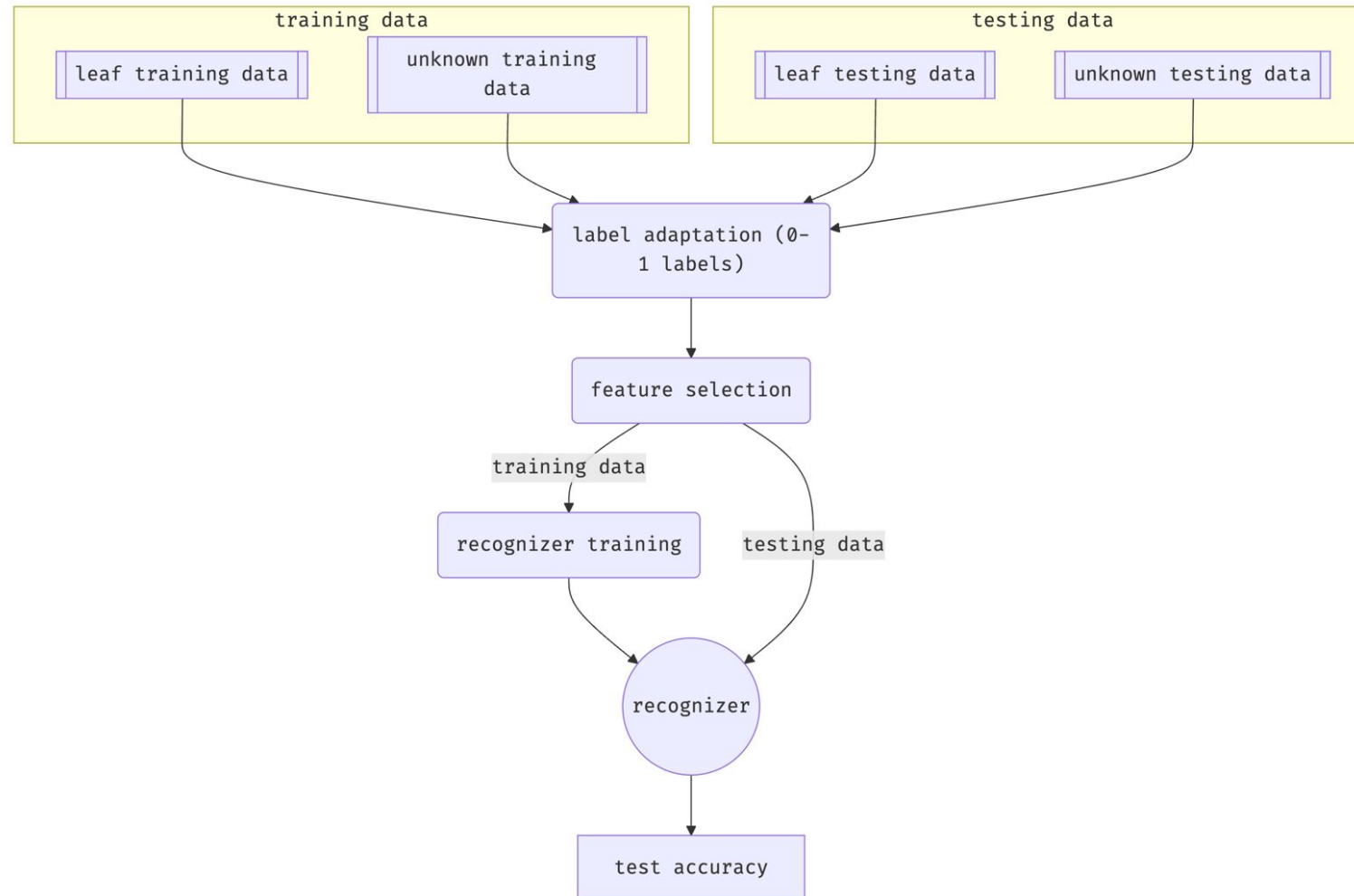
Numero di feature selezionate per il riconoscitore

In questo caso, data la scelta di un riconoscitore ensemble, è stata usata la metrica oobPermutetImportance per la selezione delle features Migliori

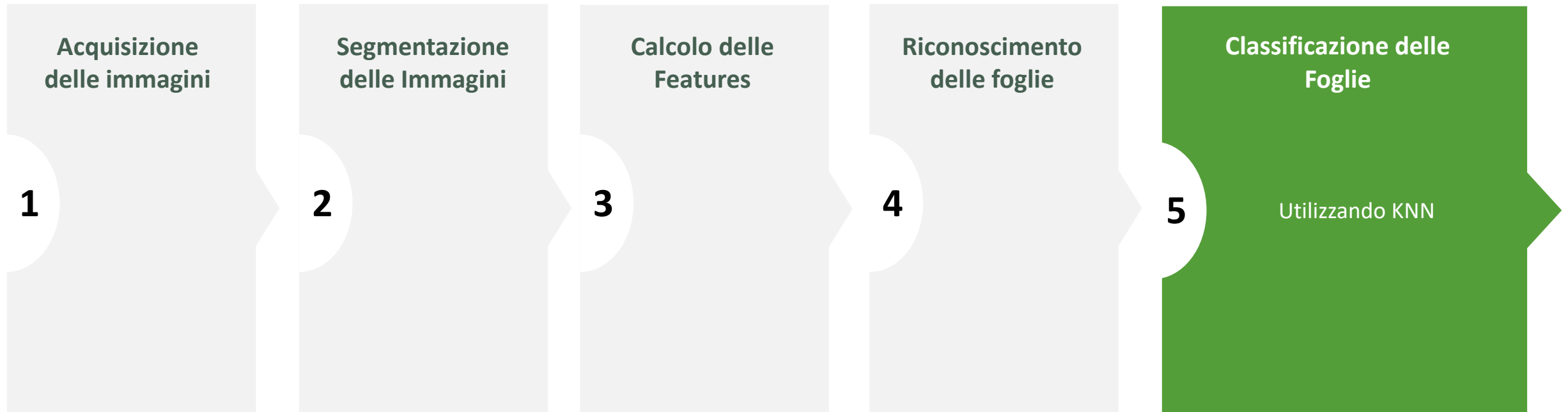
$$N \in [22, 24]$$



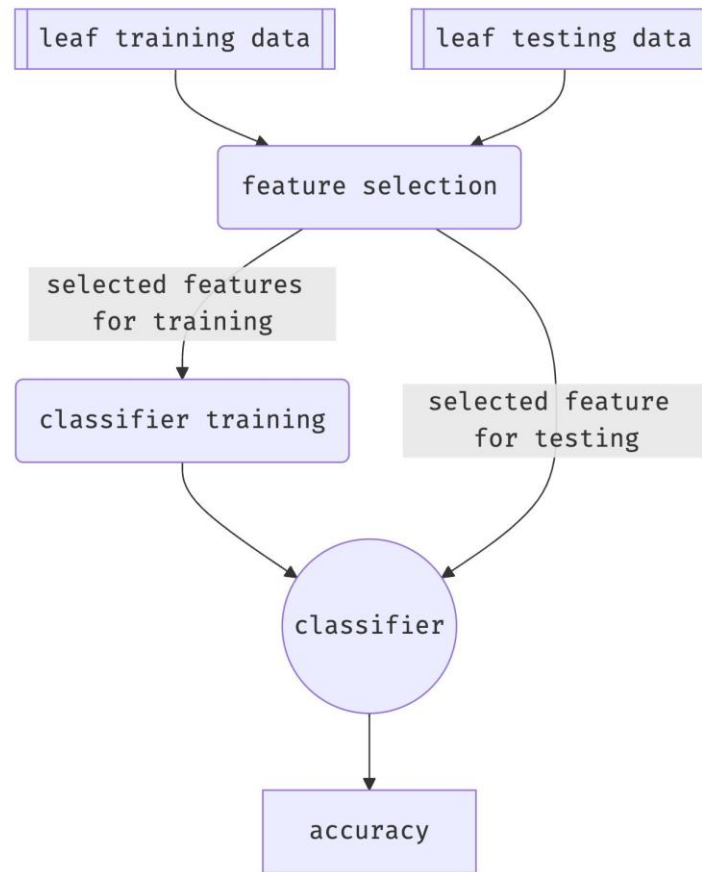
Pipeline di riconoscimento



05-Classificazione delle Foglie



Pipeline di Classificazione

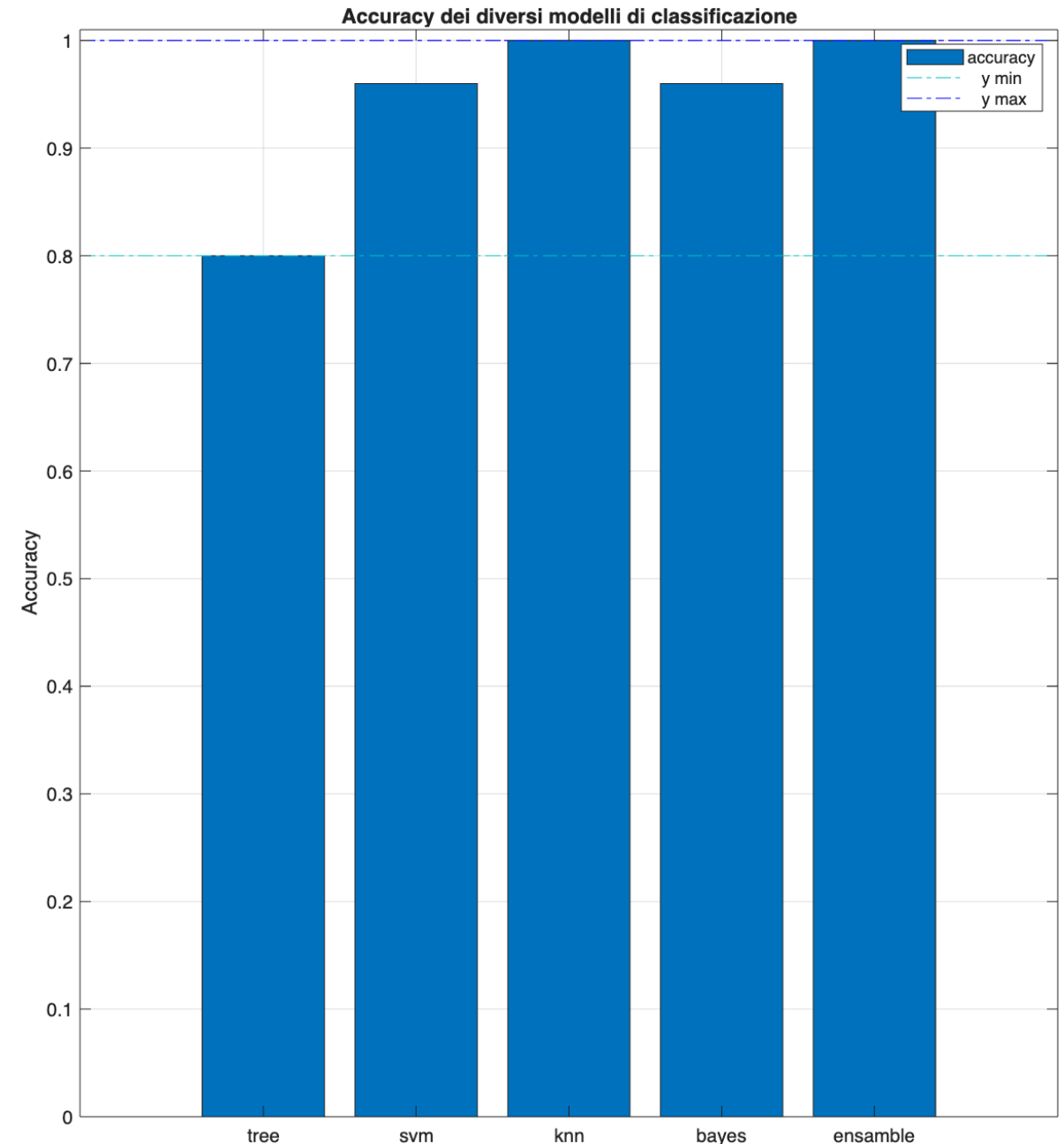


Perché KNN come classificatore?

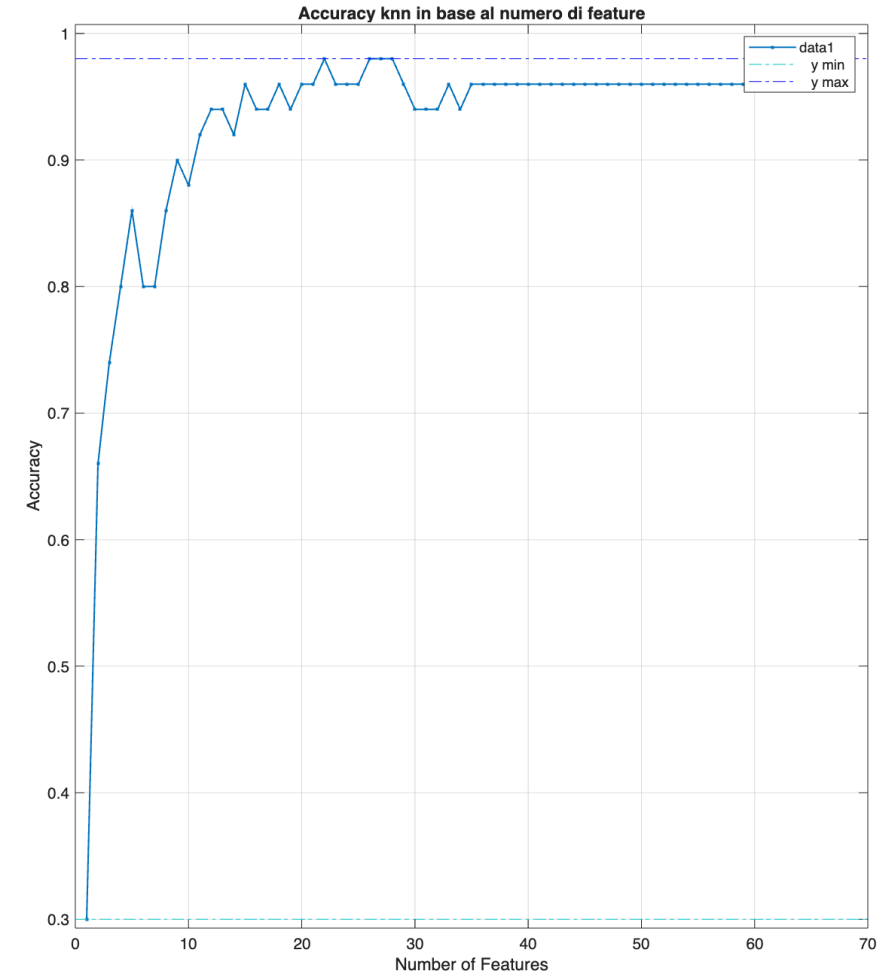
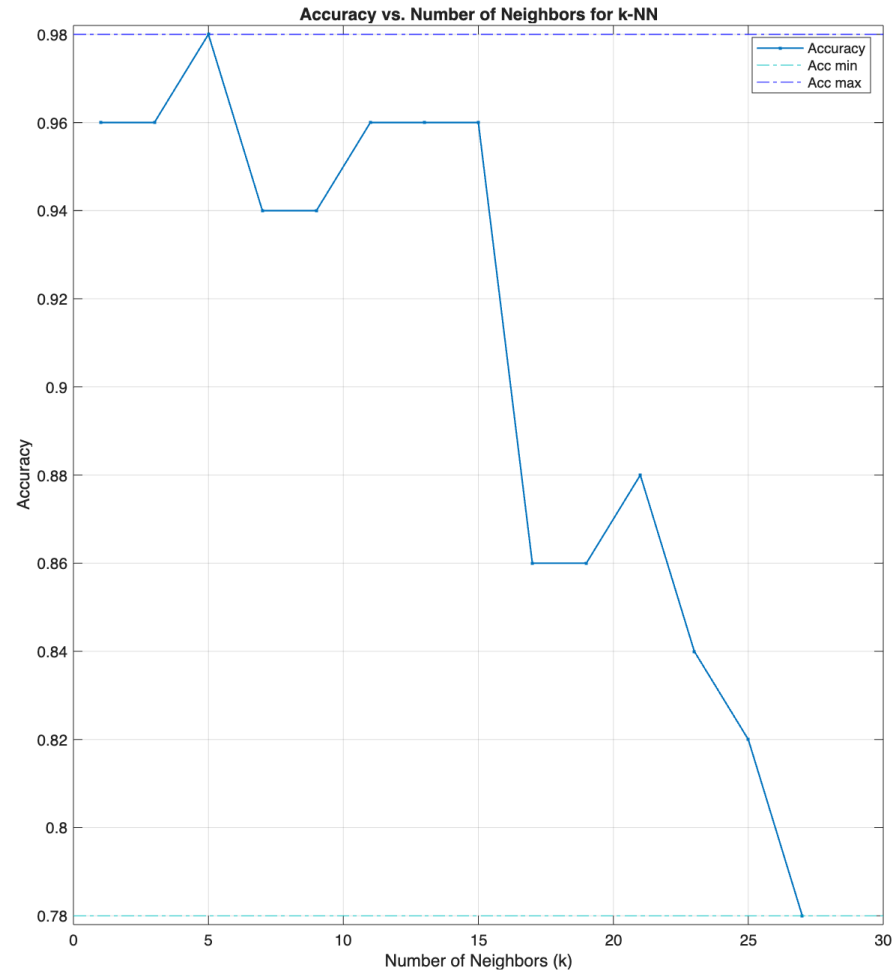
Oltre a fornire i risultati migliori sull'insieme di testing, KNN, non avendo bisogno di training, risulta più flessibile ed estensibile rispetto ad altri classificatori.

Risulta più semplice, per esempio, l'aggiunta di nuove eventuali classi.

Date le dimensioni ridotte, i tempi di classificazione rimangono contenuti. Si rende necessario, tuttavia, l'utilizzo di un riconoscitore per la gestione degli oggetti unknown, data la scarsa resistenza agli outliers di KNN.



Number of neighbours e numero di features selezionate



Segmentazione: Accuracy di **0.99** sull'interezza del dataset

Riconoscimento: Accuracy di **1** sul test set

Classificazione: Accuracy di **0.98** sul test set

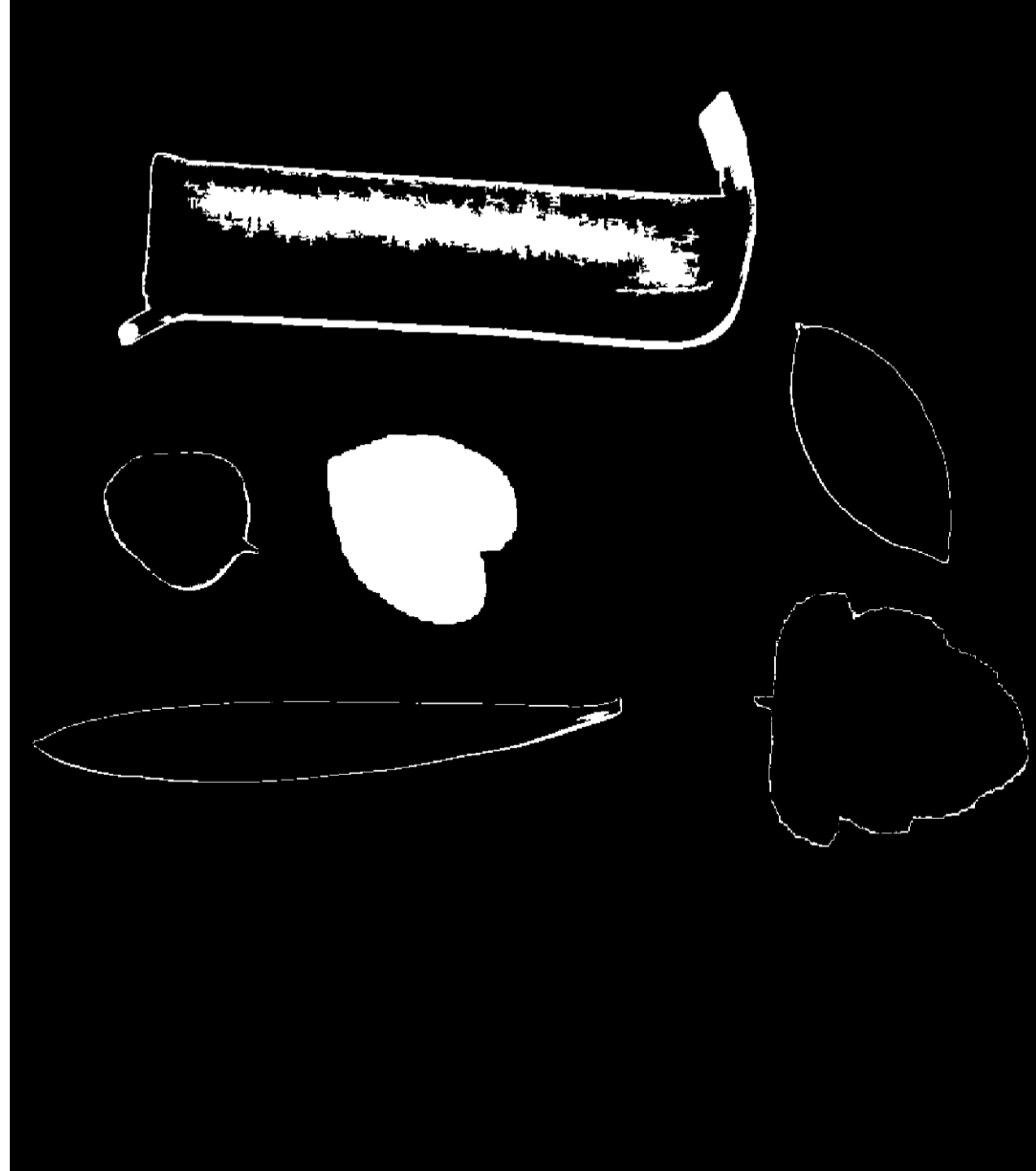


Set di immagini con **foglie miste ruotate**:

- maschere **GT**, **pixel** based accuracy: **0.78**
- maschere **GT**, **instance** based accuracy: **0.765**
- segmentazione **region growing**, **pixel** based accuracy: **0.73**
- segmentazione **region growing**, **instance** based accuracy: **0.74**

Set di immagini con foglie ruotate e oggetti casuali:

- maschere **GT**, **pixel** based accuracy: **0.56**
- maschere **GT**, **instance** based accuracy: **0.68**
- segmentazione **region growing**, **pixel** based accuracy: **0.51**
- segmentazione **region growing**, **instance** based accuracy: **0.60**



Come mai un calo così drastico?

- Le foglie del test set vengono selezionate dalle immagini che contengono anche le foglie utilizzate per il training, quindi le condizioni di luce e colore sono identiche
- Nelle immagini aggiuntive, le foglie sono ruotate e le condizioni di luce possono cambiare, inoltre le dimensioni e la distanza potrebbero variare rispetto ai dati di testing
- Nonostante siano state selezionate features fortemente invarianti a scala e rotazione, spesso la Teoria si discosta dalla pratica...
- Sulle immagini viene effettuato preprocessing prima dell'estrazione, tuttavia l'illuminazione potrebbe comunque non rimanere costante nelle diverse immagini
 - É importante anche sottolineare che il bilanciamento del bianco è basato su meccanismo white-world, quindi influenzato pesantemente dagli oggetti in scena
- Possibili errori di segmentazione



Come si potrebbe migliorare il Progetto?

- **Dataset augmentation**, con modifiche dal punto di vista di color correction, esposizione, scala e rotazione, per migliorare l'invarianza
- Meccanismo di white balance basato su elementi costanti
- Selezionare **manualmente** il **subset** di features utilizzato per il training dei modelli, basandosi sui risultati della selezione automatica, in modo da ridurre i tempi di calcolo
- Migliorare le **implementazioni manuali** (region-growing, momenti di Hu, ecc.)
- Migliore utilizzo del **calcolo parallelo** (per ora sfruttato solo in fase di estrazione delle features)
- Segmentazione con sfondi **non omogenei**

Difficoltà riscontrate durante il progetto

Inizialmente l'idea sarebbe stata quella di lavorare con dati di training e testing estratti da immagini contenenti foglie singole, per poi testare il Sistema su immagini contenenti più oggetti.

Tuttavia i risultati si sono rivelati deludenti, mostrando gli stessi problemi di calo dell'accuracy descritti in precedenza, ma di scala maggiore (accuracy < 0.4).

Per tale motivo si è deciso di estrarre i dati da immagini contenenti tutte e 15 le foglie raccolte per le classi e sfruttando delle struct per gestire la separazione tra insieme di training e insieme di testing.

Le immagini delle singole foglie sono state comunque conservate nel dataset per possibili sviluppi futuri.

Grazie per l'attenzione, buona giornata

Un progetto di:

Teodori Alessandro, 899894

e

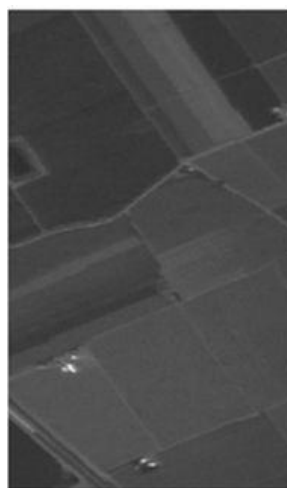
Rossi Andrea 899925



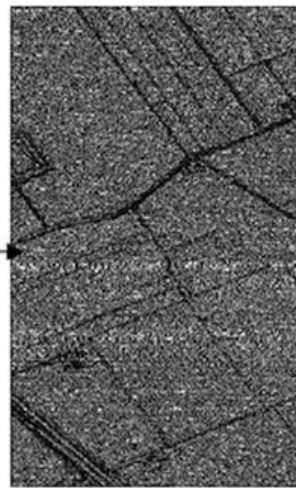
Descrittori di feature di Texture

RILBP

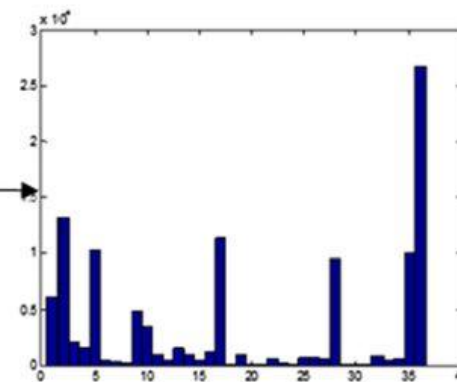
Potente descrittore di texture
dal costo computazionale
contenuto



Single Band Image
(512 × 217)
(a)



RILBP Image
(512 × 217)
(b)



RILBP Histogram
(c)

È progettato per essere invariante alla
rotazione.

Ogni pixel viene confrontato con quelli
attorno e aggiunto a una stringa
binaria che viene traslata in modo da
partire dal valore più basso.

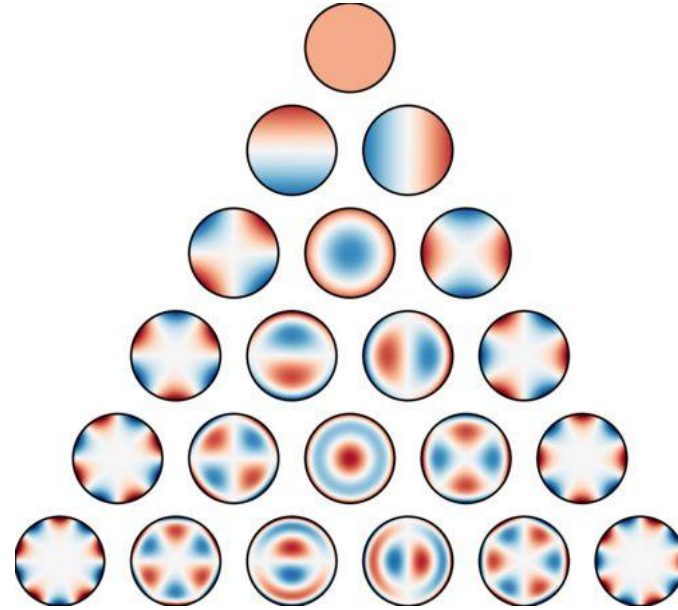
In questo modo la stessa texture produrrà lo
stesso risultato indipendentemente dalla
rotazione dell'oggetto.
Andando così a risolvere il problema della
rotazione.

Descrittori di feature di Texture

Momenti di Zernike

Descrittori basati su polinomi ortogonali definiti all'interno di un disco unitario

Sono fortemente invarianti alla rotazione e forniscono una descrizione compatta della texture

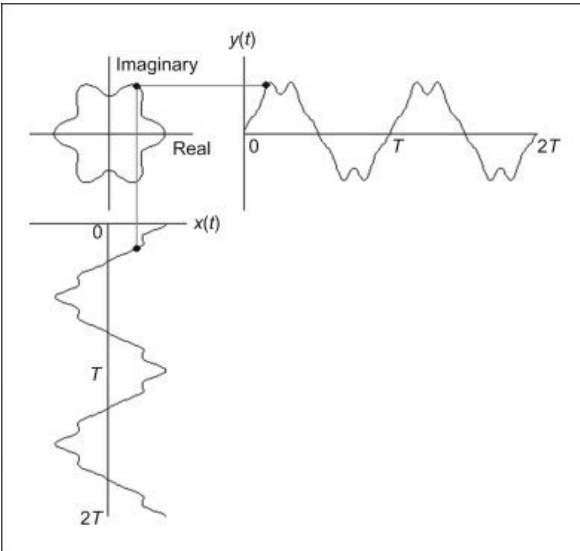


Ogni momento cattura specifiche caratteristiche della geometria, al salire dei gradi le caratteristiche si fanno sempre più fini

Descrittori di feature di Forma/Edge

Descrittori di Fourier

Permettono di rappresentare il contorno di una regione dal punto di vista del dominio delle frequenze



Il contorno viene convertito in una sequenza di numeri complessi, rendendo possibile un'analisi compatta ed efficiente tramite la trasformata di Fourier

Se l'edge signature è estratta in maniera adeguata permettono di ricavare features fortemente invarianti a scala e rotazione

RFE Loop

Si tratta di una tecnica iterativa di selezione delle features- un modello viene allenato, le features vengono valutate (MRMR o oobPermutedImportanca, in base al modello) e la *meno importante* viene eliminata

il loop continua fino al raggiungimento di un numero minimo di features (nel nostro caso viene fornito un range) o al raggiungimento di una soglia minima di importanza

Permette di ridurre la pesantezza dei calcoli, riducendo la dimensionalità del vettore features e permette al modello di generalizzare meglio, riducendo l'overfitting

Maximum Relevance Minimum Redundance(MRMR)

Maximum relevance:

Massimizza rilevanza rispetto alle classi (simile a Fisher score)

Minimum redundancy: minimizza la

ridondanza tra le feature selezionate

evitando che più feature portino la stessa informazione

Tecnica che mira a selezionare il sottoinsieme di caratteristiche più informative per la classificazione

Out-of-Bag Permuted Importance (OOBPermutedImportance)

Permette di stimare l'importanza delle variabili in modo affidabile nel modello Random Forest.

Usa l'errore «Out-Of-bag» e permuta ogni variabile predittiva per misurarne l'impatto sull'errore.

Di conseguenza più l'errore aumenta, più la variabile è importante.

Questa misura è utile per identificare le feature più rilevanti nel modello e migliorarne l'interpretabilità