

Università degli Studi di Napoli Federico II



Facoltà di Scienze Fisiche, Matematiche e Naturali
Corso di Laurea in Scienze e Tecnologie Informatiche

Tesi di Laurea Triennale Sperimentale

NEXt (NEural Extractor) II: un segmentatore di immagini astronomiche.

Relatori

Prof. Francesco Isgrò

Prof. Giuseppe Longo

Candidato

Civita Vellucci, matr. 566/327

Anno Accademico 2008/2009

Indice generale

Il problema.....	6
Introduzione.....	6
La pre-riduzione	9
Segmentazione.....	11
La Valutazione degli Oggetti.....	12
La Classificazione Stella/Galassia.....	13
DaME.....	14
Riferimenti Bibliografici.....	15
Siti Internet di Riferimento.....	16
La segmentazione di un'immagine astronomica	18
Funzionalità richieste per NExt2IS.....	18
Definizione formale di segmentazione.....	20
Reti Neurali Non Supervisionate.....	20
Il Clustering dei pixel dell'immagine.....	22
Spazio di input e costruzione del dataset.....	22
Riduzione della dimensionalità.....	24
Reti neurali per l'estrazione delle caratteristiche.....	24
La PCA.....	25
Reti neurali PCA.....	26
PCANN Standard.....	28
PCANN Non Standard.....	29
PCANN Robusta.....	30
PCANN Non Lineari.....	30
Reti neurali non supervisionate per il Clustering.....	30
Reti neurali non supervisionate per il clustering a morfologia fissa.....	31
Reti neurali KMeans.....	34
Reti Neurali Maximum Entropy.....	34
Reti Neurali Neural Gas.....	35
Reti neurali SOM.....	36
Reti neurali non supervisionate per il clustering a morfologia variabile.....	37
Reti neurali GCS.....	38
Reti neurali non supervisionate multi-livello per il clustering.....	39
Estrazione delle finestre significative.....	43
Estrattore di finestre significative basato sul metodo dell'istogramma.....	45
La Costruzione della Maschera di Segmentazione.....	48
L'Ordinamento dei Pixel.....	48
La Configurazione.....	48
Il Salvataggio dei Risultati Intermedi.....	49
Esempio di segmentazione.....	50
Il Sistema NExt2IS e i suoi Sotto-Sistemi.....	50
Riferimenti Bibliografici.....	51
Siti internet di riferimento.....	52
3 Esperimenti di segmentazione.....	53
3.1 Estrattore di finestre significative.....	54
3.2 Estrazione delle caratteristiche.....	61
3.3 Pattern Clustering.....	63
3.1.1 KMEansClusteringNN.....	63
3.1.2 MaximumEntropyClusteringNN.....	64
3.1.3 NeuralGasClusteringNN.....	65
3.1.4 SOMClusteringNN.....	66
3.1.5 GCSClusteringNN.....	67

4 Conclusioni e sviluppi futuri.....	69
Appendice.....	70
Ringraziamenti.....	102

1 Il problema

1.1 Introduzione

Da oltre un decennio l'evoluzione tecnologica dei telescopi e dei rivelatori digitali (Charge Coupled Devices o CCD) ha messo a disposizione della comunità astronomica una nuova generazione di strumenti, detti “telescopi di survey”, consistenti di un telescopio a grande campo corredato nel piano focale di un rivelatore panoramico formato da un mosaico di CCD di grandi dimensioni. Come si vedrà nel seguito di questo lavoro, l'analisi dei dati (prevalentemente immagini) prodotti da questi strumenti pone problemi computazionali non banali.

Innanzitutto per la mole dei dati da analizzare. Una survey, in genere, consiste nell'osservazione di un'ampia regione di cielo (alcune migliaia di gradi quadrati) in più bande fotometriche. In altre parole, ogni singola porzione di cielo viene osservata più volte, attraverso filtri con caratteristiche spettrali diverse.

Un telescopio da survey tipico produce, infatti, oltre 100 GB di immagini astronomiche di grandi dimensioni (fino a 32Kx32k pixel) per notte, ed una survey a grande campo può facilmente superare i 30 TByte di dati grezzi a cui vanno aggiunti circa 5 TB di dati di calibrazione.

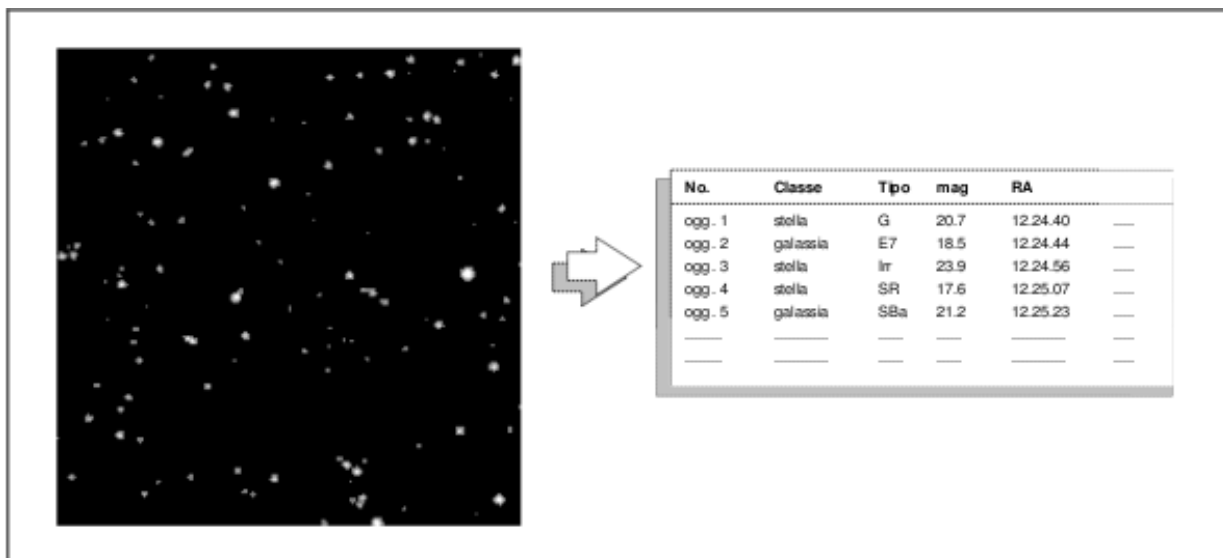


figura 1: Esempio di Catalogo Celeste

Una volta acquisiti, questi dati devono essere innanzitutto corretti per la cosiddetta “firma strumentale” o, come si dice in gergo astronomico, “pre-ridotti”, per poi essere calibrati ed analizzati. Il passo finale di quest'analisi consiste nella segmentazione dell'immagine, di cui si dirà tra breve, e nell'estrazione dei cataloghi finali: tabelle contenenti, oltre ad un identificatore univoco per ogni oggetto presente nel campo osservato, anche una lunga ed serie di parametri misurati a partire dall'immagine originale.

In genere la maggior parte degli utenti finali non vuole ripetere la complessa catena di dati che porta dal dato grezzo al catalogo e, quindi, è necessario che questi cataloghi siano quanto più possibile accurati, affidabili e verificabili.

L'affidabilità di un catalogo è definita da una serie di fattori. Ai fini della presente tesi ci si limiterà

ad elencarne solo due: completezza e contaminazione.

La *completezza* è data dalla frazione di oggetti presenti nell'immagine originale che è effettivamente ritrovata nel catalogo finale. Gli oggetti astronomici hanno luminosità apparenti (flussi osservati) molto diversi l'uno dall'altro e chiaramente, gli oggetti più deboli coprono un minore numero di pixel ed hanno un rapporto segnale/rumore più basso. Ciò ovviamente ne rende difficile la rivelazione ed aumenta la probabilità che l'oggetto non venga correttamente identificato (estratto) dall'immagine. La completezza, ovviamente, diminuisce al diminuire della luminosità degli oggetti osservati.

La *contaminazione* dà invece la percentuale di oggetti spuri, cioè la frazione di oggetti ritrovati nei cataloghi che non trovano alcun riscontro nell'immagine reale. Anche in questo caso, il problema si pone soprattutto con gli oggetti deboli. E' facile infatti che un gruppo di pixel in cui è presente solo il rumore venga erroneamente classificato come oggetto a causa delle fluttuazioni statistiche del segnale.

Come si vedrà, questi due parametri dipendono quasi esclusivamente dall'algoritmo scelto per operare la segmentazione dell'immagine, cioè il riconoscimento dei pixel che contengono segnale utile da quelli che, invece, contengono solo rumore e/o fondo cielo.

Un catalogo ideale dovrebbe avere completezza 1 e contaminazione 0 a tutti i livelli di flusso ma, in pratica la completezza diminuisce e la contaminazione aumenta man mano che si prendono in considerazione oggetti sempre più deboli. Ciò fa sì che molto spesso, per rispondere a problemi scientifici diversi in cui può essere più importante ottimizzare la completezza rispetto all'efficienza o viceversa, gli astronomi siano costretti a spendere molto tempo nella caratterizzazione dei cataloghi (post processing) o, in casi estremi, a ripetere integralmente l'intera sequenza pre-riduzione, riduzione, analisi. Operazione questa che è in genere *time-consuming* e diviene sostanzialmente impossibile quando si ha a che fare con dataset di molti Tbyte.

Attualmente, la comunità astronomica opera l'estrazione dei cataloghi utilizzando il pacchetto SExtractor (SEx) scritto dal francese E. Bertin nel 1979. Il funzionamento di questo pacchetto

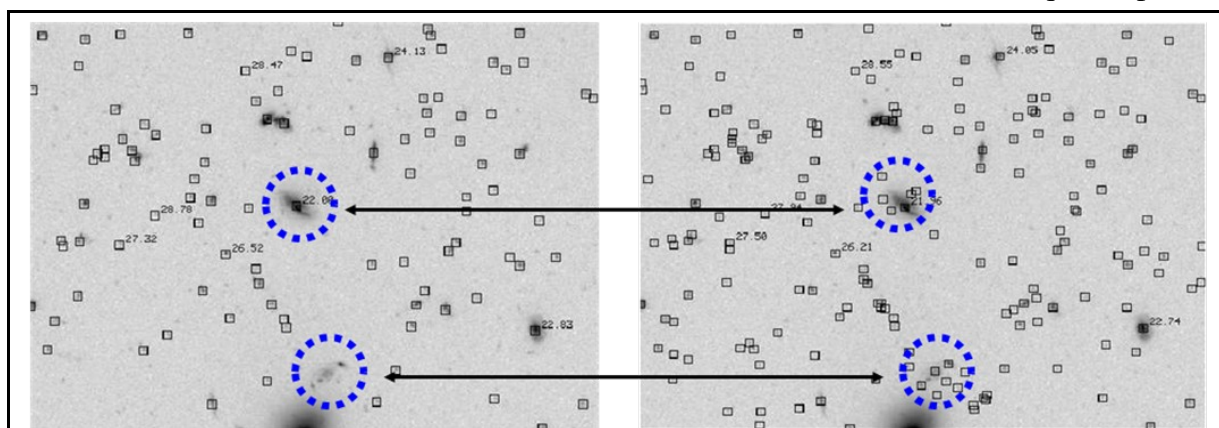


Figura 1: confronto di due diversi run di SEx su una stessa immagine astronomica. I due run sono stati ottenuti da autori diversi che hanno imposto valori diversi alla soglia di rivelazione. Gli oggetti effettivamente rivelati e presenti nei cataloghi finali sono indicati dai quadratini. Si notino le differenti completezza e contaminazione dei due cataloghi (da Ferguson, 1988).

richiede però la definizione da parte dell'utente di alcuni parametri arbitrari, primo tra tutti l'assegnazione di una soglia di rivelazione (espressa relativamente alla varianza del fondo cielo). Quest'arbitrarietà fa sì che cataloghi di una stessa immagine estratti da utenti diversi possano condurre a risultati sostanzialmente diversi (vedi, Fig.1).

1 Il problema

Il problema è reso ancor più complesso da altri fattori. Per meglio spiegarli occorre una breve premessa. Da un punto di vista osservativo gli oggetti astronomici possono essere distinti in due grandi categorie: stelle e galassie. Questa terminologia può ingenerare confusione in quanto quello che realmente si intende è distinguere distinzione tra oggetti non risolti ed oggetti risolti.

Gli oggetti non risolti (stelle), sono oggetti puntiformi cioè oggetti la cui forma è indistinguibile dalla Point Spread Function (PSF) strumentale. In altre parole, essi appaiono come figure di diffrazione (di Airy) di sorgenti puntiformi. Gli oggetti risolti (galassie), sono invece oggetti la cui forma è sostanzialmente diversa da quella della PSF e che appaiono più o meno estesi nelle immagini. In un qualsiasi campo celeste coesistono sia oggetti puntiformi che oggetti estesi e molto spesso gli oggetti puntiformi appaiono sovrapposti a quelli estesi.

Dal punto di vista dei cataloghi finali, questo in genere, influisce sulla completezza e sulla contaminazione rendendo entrambe una funzione piuttosto complessa della posizione all'interno dell'immagine. Si prenda, per esempio, il caso in cui la rivelazione viene effettuata utilizzando un criterio di soglia definito in funzione del valore del fondo cielo.

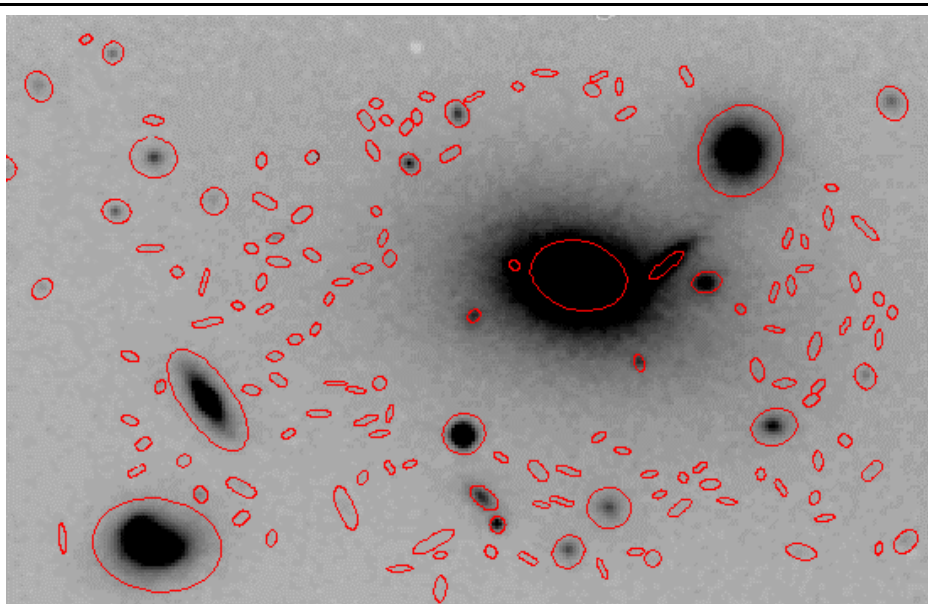


Figura 2: esempio di come la presenza di un oggetto esteso comporti un aumento della rivelazione di oggetti spuri nelle sue regioni più esterne.

Nelle regioni periferiche di un oggetto esteso, l'innalzamento del valore di fondo fa sì che molti picchi di rumore vadano al di sopra della soglia e generino una rivelazione spuria (aumento della contaminazione).

Nelle regioni più interne e più brillanti, invece, il contributo degli oggetti puntiformi può divenire trascurabile e, quindi portare ad una non rivelazione e ad una diminuzione della completezza (Ved. Figura 2).

Un ulteriore fattore da tenere presente è di natura più sottile. Come si è detto, le immagini astronomiche vengono in genere acquisite in più bande fotometriche. A parità di tutti i parametri osservativi (tempo d'esposizione, etc.) , le diverse trasmittanze dei filtri (bande) e le diverse sensibilità spettrali del rivelatore fanno sì che il flusso limite rilevabile vari da banda a banda. A ciò occorre aggiungere il fatto che gli stessi oggetti celesti hanno luminosità intrinseche diverse nelle diverse bande e quindi oggetti che appaiono brillanti in una data banda possono essere molto deboli

o addirittura invisibili in un'altra banda.

Nella misurazione dei parametri è quindi molto importante tenere traccia di ciò che accade nelle diverse bande fotometriche. Si prenda ad esempio il caso in cui su un certo pixel, in una certa banda si riesce a riconoscere un oggetto mentre in un'altra banda l'unico segnale che si registra in corrispondenza dello stesso pixel è quello del fondo cielo. Nel misurare i parametri occorre ricordarsi che, in questo caso, il pixel della seconda banda non corrisponde a quello che in genere verrebbe chiamato fondo cielo, ma piuttosto ad una non rilevamento che può essere usata per ricavare un limite superiore al flusso emesso dall'oggetto.

1.1 La pre-riduzione

Per meglio comprendere quali sono i requisiti da soddisfare affinché i cataloghi finali siano di qualità soddisfacente è opportuno soffermarsi, seppure brevemente, su le varie fasi della catena di riduzione.

Le immagini astronomiche sono matrici di pixel e sono caratterizzate da un numero ridotto di parametri:

- dimensione del pixel. In genere compresa tra 22 e 35 micron. Insieme alla lunghezza focale del telescopio determina la cosiddetta “scala”: l'angolo solido coperto da ogni pixel sulla sfera celeste che in genere si esprime in arsec/pixel;
- numero dei pixel ($m \times n$)
- Intervallo dinamico: determinato dall'elettronica di acquisizione dà il valore minimo e massimo del segnale registrabile su ogni singolo pixel. Nei moderni CCD astronomici l'intervallo dinamico è tipicamente [0-65532]

Le immagini astronomiche sono in genere in formato FITS. Il FITS prevede un'intestazione contenente una serie di metadati che consentono di ricostruire la configurazione strumentale al momento dell'acquisizione dell'immagine: posizione sulla volta celeste, epoca di acquisizione, tempo di esposizione, tipo di filtro inserito lungo il percorso ottico (in gergo astronomico detta anche “banda fotometrica”), etc. Una conoscenza accurata di questi parametri è in genere indispensabile per consentire una corretta riduzione ed analisi dei dati.

In prima approssimazione (più che sufficiente ai fini del presente lavoro), il segnale S_{ij} registrato nel pixel di coordinate (i,j) può essere rappresentato come:

$$S_{ij}(\lambda) = \alpha_{ij}(\lambda)(F(\lambda) + G_{ij}) + B_{ij} + D_{ij} + N_{ij}$$

Dove:

- $F(\lambda)$ è il segnale proveniente dall'oggetto osservato ed è la quantità che si vuole misurare.
- G_{ij} è un segnale spurio ineliminabile, comunemente detto “fondo cielo”. Deriva da un complesso di fattori quali, ad esempio, fotoni diffranti dai granuli di polvere presenti nell'atmosfera, fotoni emessi dalla ricombinazione di ioni atmosferici, luce proveniente da oggetti deboli di sfondo non risolti e non rivelabili individualmente, etc.
- α_{ij} è la funzione di trasferimento del singolo pixel. Nei moderni rivelatori è in genere lineare sulla quasi totalità dell'intervallo dinamico.
- B_{ij} è il cosiddetto bias o “fat zero”. Un segnale di ampiezza costante che si aggiunge prima

1 Il problema

di ogni esposizione in modo da facilitare la lettura dei segnali deboli.

- D_{ij} è il contributo detto “corrente di oscurità”, generato dal rumore termico
- N_{ij} è il rumore (in genere poissoniano) legato alla statistica dei fotoni incidenti

α_{ij} , B_{ij} e D_{ij} determinano la cosiddetta firma strumentale e devono in qualche modo essere calcolati e se possibile rimossi durante la fase di preriduzione.

Ciò è reso possibile tramite l’acquisizione di immagini (frame) di calibrazione detti, rispettivamente, *flat field*, *bias* e *dark*.

- Flat field: è un’immagine ottenuta esponendo il telescopio, nella stessa configurazione usata per l’acquisizione delle immagini scientifiche, ad un campo di radiazione uniforme. La variazione del segnale registrata nei diversi pixel può quindi essere ricondotta ad una variazione della pendenza della curva di trasferimento e, quindi usata per stimare in modo accurato α_{ij}
- Bias: viene ottenuto esponendo per tempo nullo il CCD mantenendo l’otturatore della camera chiuso. Essendo il tempo di esposizione nullo, il contributo del dark è zero e quindi l’immagine può essere utilizzata per correggere il termine B_{ij}
- Dark: si ottiene esponendo il rivelatore mantenendo l’otturatore della camera chiuso, per un tempo confrontabile con quello delle esposizioni scientifiche. Una volta sottratto il bias, ciò che resta è la corrente di oscurità.

Al termine della procedura di pre-riduzione, si ottiene il valore

$$(S_{ij}(\lambda) - B_{ij} - D_{ij}) / \alpha_{ij}(\lambda) = F(\lambda) + G_{ij} + N_{ij}/\alpha_{ij}(\lambda)$$

Dove $F(\lambda)$ è il segnale ed il termine $G_{ij} + N_{ij}/\alpha_{ij}(\lambda)$ il rumore.

La segmentazione di un’immagine astronomica consiste nel riconoscere i pixel per cui il segnale è dominante rispetto al rumore.

Il presente lavoro è consistito nel mettere a punto una procedura – basata per lo più su tecniche neurali – per la classificazione dei pixel in presenza di un insieme minimale di parametri arbitrari o di ipotesi a priori. Il lavoro rappresenta un’evoluzione (in termini di maggiore automazione, scalabilità e affidabilità) del software NExt (Neural Extractor) descritto in [Andreoni et al. 2001].

Le caratteristiche salienti di NExt sono le seguenti:

- (i) non richiede alcuna definizione a priori di cosa sia un oggetto, ma soltanto la definizione di adiacenza tra pixel;
- (ii) non richiede alcuna impostazione particolare per il rilevamento e classificazione degli oggetti presenti sull’immagine;
- (iii) per eseguire la classificazione stella/galassia, non fa uso di un insieme arbitrario di caratteristiche, ma seleziona in modo oggettivo le più significative.

La strategia adottata da NExt per elaborare un’immagine astronomica (schematizzata in figura 2) si articola in tre fasi:

1. segmentazione dell’immagine di input;
2. valutazione e riconoscimento degli oggetti presenti sull’immagine (deblending): misurazione

dei parametri astronomici;

3. classificazione di ciascuno degli oggetti presenti sull'immagine come stella, o come galassia.

Delle tre fasi sopraelencate, verrà trattata dettagliatamente solo la prima, in quanto essa costituisce l'obiettivo del lavoro svolto. Tale lavoro è consistito in:

- (I) reverse-engineering del software di segmentazione pre-esistente, implementato in C/C++;
- (II) riprogettazione (e, conseguente re-implementazione in C++), secondo il paradigma Object Oriented (OO) finalizzata a:
 - migliorare robustezza ed affidabilità;
 - assicurare la riusabilità della maggior parte dei componenti software (quali le reti neurali non supervisionate e gli estrattori di finestre, descritti nel capitolo successivo);
 - assicurare la facilità d'uso da parte del Client (applicazioni esterne, o parti di esse).

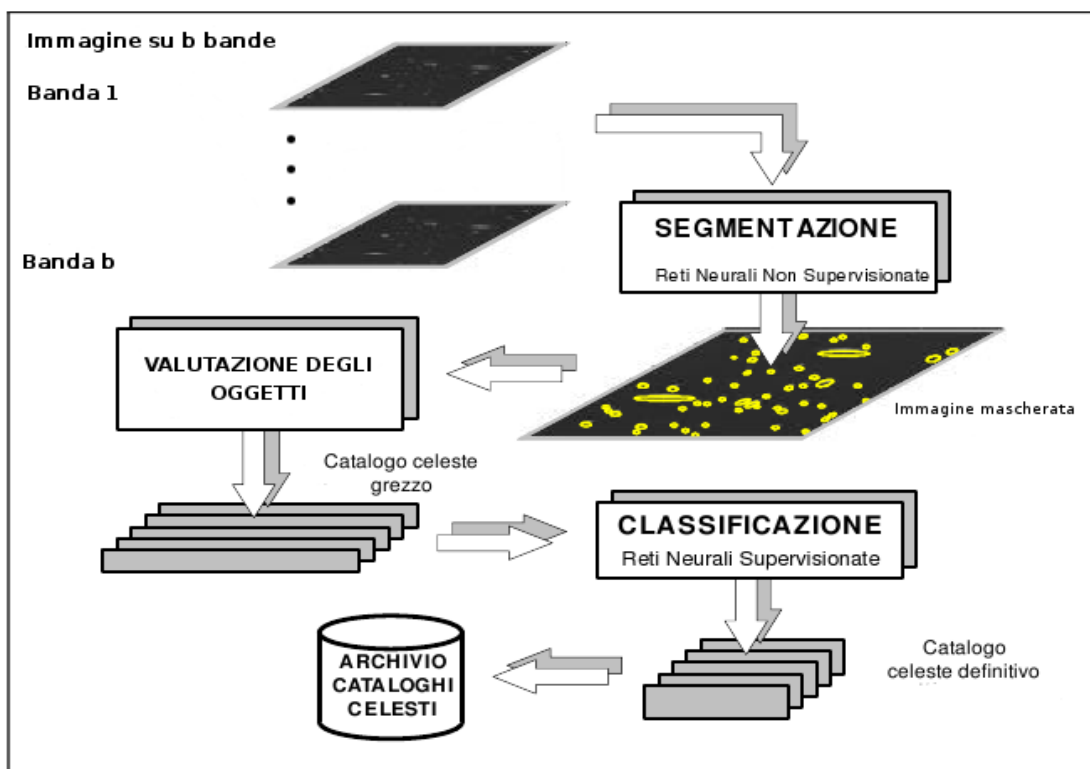


figura 2: Architettura NExt

Per quanto riguarda le fasi 2 e 3 ne verrà fornita, per completezza, solo una descrizione illustrativa.

1.1.1 Segmentazione

La segmentazione è un processo di ripartizione di un'immagine in regioni connesse e vicendevolmente disgiunte, dove ogni regione è omogenea e l'unione di regioni adiacenti è non omogenea.

Per un'immagine astronomica, la segmentazione è il mezzo tramite il quale si individuano tutti i

corpi celesti in essa contenuti, evidenziandone le caratteristiche in modo più netto e filtrando i contributi relativi al rumore. Pertanto, tra tutte le tecniche di segmentazione presenti in letteratura, si è scelto di adottare quelle basate sull'utilizzo delle reti neurali non supervisionate, in quanto esse appaiono offrire prestazioni migliori in ambienti rumorosi [Bishop, Rojàs]

In tal caso, la segmentazione diviene a tutti gli effetti un problema di classificazione di ciascun pixel, interno all'immagine, come appartenente ad un oggetto oppure allo sfondo. Il risultato di tale classificazione è una *maschera* dell'immagine di input, che indica quali pixel di quest'ultima costituiscono un oggetto e quali no.

La fase di segmentazione di NExtII si articola in due sotto-fasi principali:

- 1.il clustering dell'insieme dei pixel dell'immagine e la costruzione della *maschera di segmentazione* dell'immagine di input;
- 2.l'ordinamento dei pixel dell'immagine in base alla luminosità del cluster di appartenenza.

La fase di segmentazione di Next2 è schematizzata in figura 3 e descritta più dettagliatamente nel Capitolo successivo.

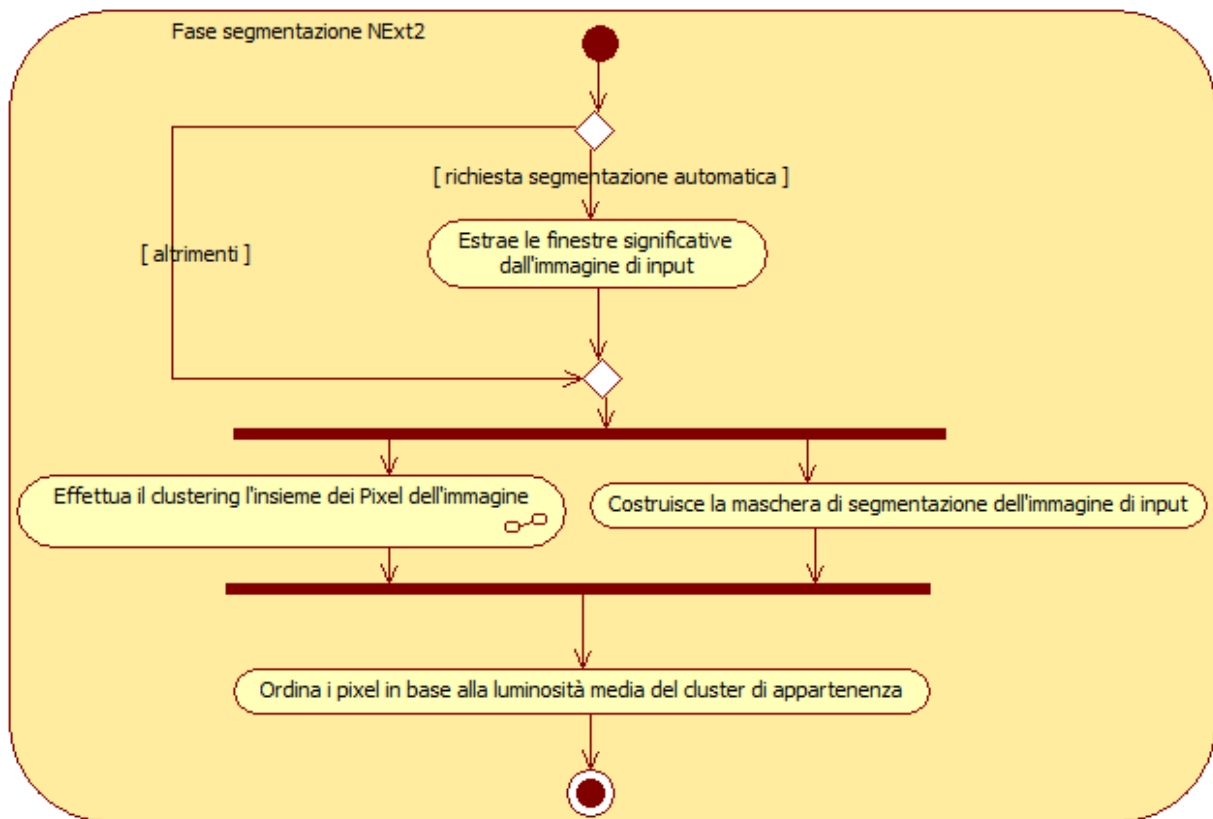


figura 3: Diagramma, semplificato, delle Attività per la fase di segmentazione di NExt2

1.1.2La Valutazione degli Oggetti

La successiva fase di NExt è, come precedentemente accennato, la valutazione degli oggetti individuati nella fase di segmentazione. La fase di valutazione degli oggetti consiste nel misurare, per ciascun oggetto, un insieme di parametri fotometrici. L'input di tale fase è costituito

dall'immagine di input e dalla relativa *maschera di segmentazione*; il suo output è, invece, un file rappresentante il catalogo *grezzo* dei corpi celesti presenti nell'immagine di input. In tale file, ogni riga rappresenta un oggetto, e ogni colonna rappresenta il risultato della misurazione di una determinata caratteristica (parametro fotometrico) per quell'oggetto.

La fase di valutazione degli oggetti (schematizzata in figura 4) si suddivide in quattro fasi:

- 1.rilevamento degli oggetti presenti sull'immagine mascherata, cioè il risultato dell'applicazione all'immagine di input della propria *maschera di segmentazione*;
- 2.separazione di oggetti multipli: un oggetto multiplo rappresenta più di un corpo celeste. Tali oggetti si generano quando più corpi celesti si trovano talmente vicini (sull'immagine) da non essere distinguibili in fase di segmentazione;
- 3.raffinamento dei contorni degli oggetti rilevati: la segmentazione crea spesso forme non perfettamente corrispondenti a quelle dei corpi celesti che vuole rappresentare. In tale sotto-fase, si deve attenuare questo sfasamento cercando di far corrispondere i centri e di coprire l'intera area informativa;
- 4.estrazione delle caratteristiche, e costruzione del catalogo *grezzo* degli oggetti rilevati.

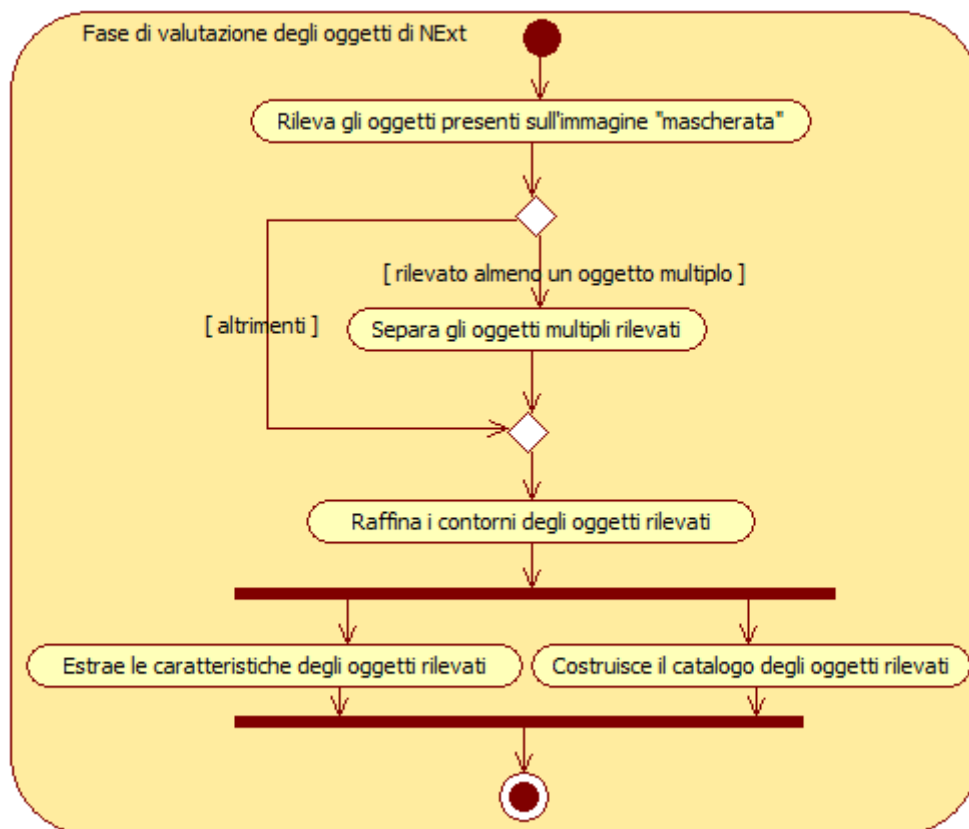


figura 4: Diagramma, semplificato, delle Attività per la fase di valutazione degli oggetti di NExt

1.1.3 La Classificazione Stella/Galassia

Nella fase di classificazione stella/galassia di NExt si sviluppano i criteri per la separazione tra stelle e galassie. A tale scopo sono state impiegate metodologie neurali supervisionate (MLP – Multi-Layer Perceptron). Ogni oggetto rilevato nella precedente fase di valutazione, viene

classificato come stella, o come galassia, Solo al termine di questa fase sarà possibile studiare i corpi celesti presenti sull'immagine.

L'input di questa fase è costituito dal solo catalogo *grezzo*, ottenuto in fase di valutazione degli oggetti; ed il suo output è il catalogo completo dei corpi celesti presenti sull'immagine, ottenuto aggiungendo a quello *grezzo* di input una colonna, che rappresenta la classe (S = Stella, G = Galassia) di appartenenza di ciascun oggetto.

La fase di classificazione stella/galassia (schematizzata in figura 5) si articola in due sotto-fasi:

1. selezione delle caratteristiche da usare per la classificazione degli oggetti;
2. classificazione stella/galassia di ciascun oggetto, e conseguente raffinamento del catalogo di input.

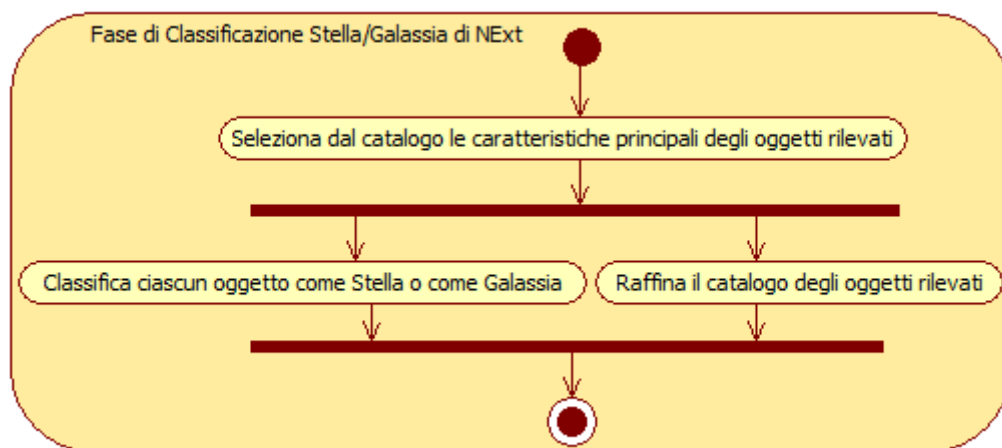


figura 5: Diagramma, semplificato, delle Attività per la fase di classificazione Stella/Galassia di NExt

1.2 DaME

Il presente lavoro è stato svolto all'interno del progetto DaME con l'obiettivo di estenderne le funzionalità. Come si vedrà nel seguito quest'obiettivo non è stato raggiunto solo attraverso lo sviluppo di una complessa applicazione per la segmentazione automatica delle immagini astronomiche, ma anche l'arricchimento delle funzionalità di base di DaME, in particolare per quanto concerne gli algoritmi neurali non supervisionati (vedi Cap.2).

DaME (Data Mining and Exploration) è un progetto congiunto del Dipartimento di Scienze Fisiche della Federico II e del dipartimento di Astrofisica del California Institute of Technology per lo sviluppo di una piattaforma di Data Mining orientata all'astrofisica, ma utilizzabile anche in altri ambiti scientifici. Il progetto è cofinanziato dal Ministero per gli Affari Esteri nell'ambito degli accordi bilaterali Italia-Stati Uniti.

L'obiettivo principale di DaME è di proporre una nuova piattaforma orientata ai servizi (Service Oriented Architecture, SOA) per lo sviluppo e la condivisione di strumenti avanzati per l'esplorazione di database scientifici di grandi dimensioni e di applicazioni di Data Mining (estrazione di conoscenza da grandi database) basate perlopiù su tecniche neurali.

Dal punto di vista scientifico il progetto DaME emerge in ambito astrofisico, laddove i principali requisiti possono essere sintetizzati nel seguente modo:

- la necessità di "federare" osservazioni reali o simulate raccolte dagli archivi mondiali, definendo una serie di standard per i loro formati e per i protocolli di accesso.
- l'implementazione di innovativi strumenti di calcolo per l'esplorazione di dati e l'estrazione di conoscenza da essi. Questi strumenti devono essere scalabili, quanto più possibile automatici e di semplice utilizzo per l'utente.

Mentre il primo requisito è pienamente realizzato dall'Osservatorio Virtuale internazionale, il secondo è lo scopo fondamentale per cui è stato iniziato il progetto DaME.

In altri termini, l'enorme quantità di dati prodotta dagli strumenti di nuovissima generazione e messa a disposizione dal Virtual Observatory (VObs) richiede una nuova generazione di strumenti per l'esplorazione e l'analisi. Va sottolineato che la vastità dei database in questione non è solo nel grande numero di record (che può arrivare a svariati milioni) ma anche e soprattutto da uno spazio dei parametri osservativi che raggiunge facilmente ordini di grandezza di centinaia di dimensioni.

Inoltre, data la complessità computazionale degli algoritmi di analisi su questi database di grandi dimensioni, è necessario utilizzare piattaforme di calcolo distribuito, come il GRID computing, accedendo in maniera trasparente per l'utente alle risorse messe a disposizione dal progetto S.Co.P.E.

La piattaforma DaME (il cui prototipo è accessibile all'indirizzo <http://dame.na.infn.it>) è stata progettata con un'architettura orientata ai servizi, in modo che ciascun componente possa essere rimpiazzato o personalizzato: ad esempio, se da un lato è previsto il rilascio di una webapplication dedicata all'interazione con il sistema, l'utente può creare la propria interfaccia utente compatibile con le specifiche della piattaforma.

Allo stesso modo è possibile estendere i driver di DaME in modo da consentire l'esecuzione delle applicazioni su altre piattaforme di calcolo distribuito.

Al termine di questo lavoro DaME risulta arricchito di un'ampia classe di algoritmi neurali non supervisionati.

1.3 Riferimenti Bibliografici

Longo G., Tagliaferri R., Andreon S.

NExT (Neural Extractor): a new automated tool to extract catalogues from astronomical images

in MINING THE SKY, Proceedings of MPA/ESO/MPE Joint Astronomy Conference held in Garching,

July 31 – August 4, 2001, (pp. 379-385).

Andreon S., Gargiulo G., Longo G., Tagliaferri R. e Capuano N.

Wide Field imaging – I. Applications of neural networks to object detection and star/galaxy classification

2000 RAS, MNRAS 319, 700-716.

Capuano N.

Classificazione Stelle/Galassie tramite reti neurali.

Tesi di Laurea in Scienze dell'Informazione, presso la Facoltà di Scienze MM. FF. NN.

dell'Università degli Studi di Salerno, A. A. 1997/1998, Capitolo 1.

Rizzo E.

Un tool basato su reti neurali per il riconoscimento di oggetti astronomici

Tesi di Laurea in Informatica, presso la Facoltà di Scienze MM. FF. NN. dell'Università degli Studi di Salerno, A. A. 2004/2005, Capitolo 1.

Laurino O.

2009 "Deprojecting the AGN Universe within the Virtual Observatory".

1.4 Siti Internet di Riferimento

<http://dame.na.infn.it>

<http://voneural.na.infn.it>

<http://www.scope.unina.it>

2 La segmentazione di un'immagine astronomica

Il cuore del pacchetto NExt2 è il componente (NExt2IS) che si occupa della segmentazione delle immagini. Per segmentazione intendiamo il raggruppamento (clustering) dei pixel dell'immagine in due distinte classi: segnale e fondo. Il risultato della segmentazione è dunque una maschera binaria in cui i pixel accesi formano gli oggetti astronomici che verranno catalogati, mentre i pixel spenti rappresentano il fondo, ovvero le regioni di cielo che non contengono oggetti.

Nel caso di NExt2IS la segmentazione avviene in due *stadi*:

1. il clustering dell'insieme dei pixel dell'immagine e la costruzione della *maschera di segmentazione* dell'immagine di input;
2. la classificazione dei pixel dell'immagine, in base alla luminosità media del cluster di appartenenza.

Questo semplice schema verrà esaminato più in dettaglio nel seguito del capitolo.

NExt2 fa ampio uso di tecniche di intelligenza artificiale, in particolare di reti neurali non supervisionate, per effettuare il clustering dei pixel dell'immagine e fornire all'utente la maschera binaria rappresentante la segmentazione voluta.

Nei prossimi paragrafi verranno elencati gli algoritmi neurali usati da NExt2. Ciascuno di questi algoritmi è stato sviluppato in modo da poter funzionare in maniera del tutto indipendente dal segmentatore: le relative classi compongono una libreria software condivisa (*shared object*), invocabile da software di terze parti.

Come evidenziato nel Capitolo 1, infatti, gli algoritmi utilizzati da NExt2, sono utilizzabili in un vasto spettro di applicazioni per l'estrazione di conoscenza dai dati (Data Mining). La libreria che compone il motore di NExt2 entrerà quindi a far parte dei Data Mining Models della suite di DaME.

2.1 Funzionalità richieste per NExt2IS

La fase di segmentazione si articola nelle seguenti attività:

1. Estrazione, ad opera di un opportuno estrattore (MeaningFullWindowsExtractor), delle *finestre significative*, cioè aree quadrate dell'immagine di input da cui estrarre i Pattern per addestrare le reti non supervisionate usate per il clustering dei Pixel dell'immagine di input;
2. Riduzione della dimensionalità dello spazio degli input, ad opera di una rete non supervisionata (FeaturesExtractionNN). I Pattern vengono estratti dall'immagine di input e rappresentano, come verrà illustrato in seguito, la luminosità di un Pixel e del “contesto” in cui si trova;
3. Clustering dell'insieme dei Pattern, rappresentanti l'output dell'attività 2, ad opera di una rete neurale non supervisionata multi-livello (MultiLayerClusteringNN).

I Client di NExt2IS sono applicazioni esterne, o parte di esse, che hanno bisogno di segmentare un'immagine.

Le funzionalità offerte da NExt2IS sono le seguenti:

- configurazione, separata e non, dei componenti principali per la segmentazione (semi-)automatica di un'immagine;
- segmentazione (semi-)automatica di un'immagine, con possibilità di salvare, in appositi file, i risultati intermedi della segmentazione.

Inoltre:

- i file di output di NExt2IS, si limitano a quelli per il salvataggio dei risultati intermedi (informazioni di ripristino), della segmentazione in corso;
- i file di input di NExt2IS, si limitano agli eventuali
 - file (.fits) contenente l'immagine di input;
 - file leggibile da ds9, contenente le finestre significative dell'immagine, per la sua segmentazione semi-automatica;
 - ai file di configurazione e a quelli di ripristino;
- la MultiLayerClusteringNN usata da NExt2IS ha un numero fisso di livelli pari a 3.

Le principali funzionalità offerte da NExt2IS sono schematizzate in figura 6.

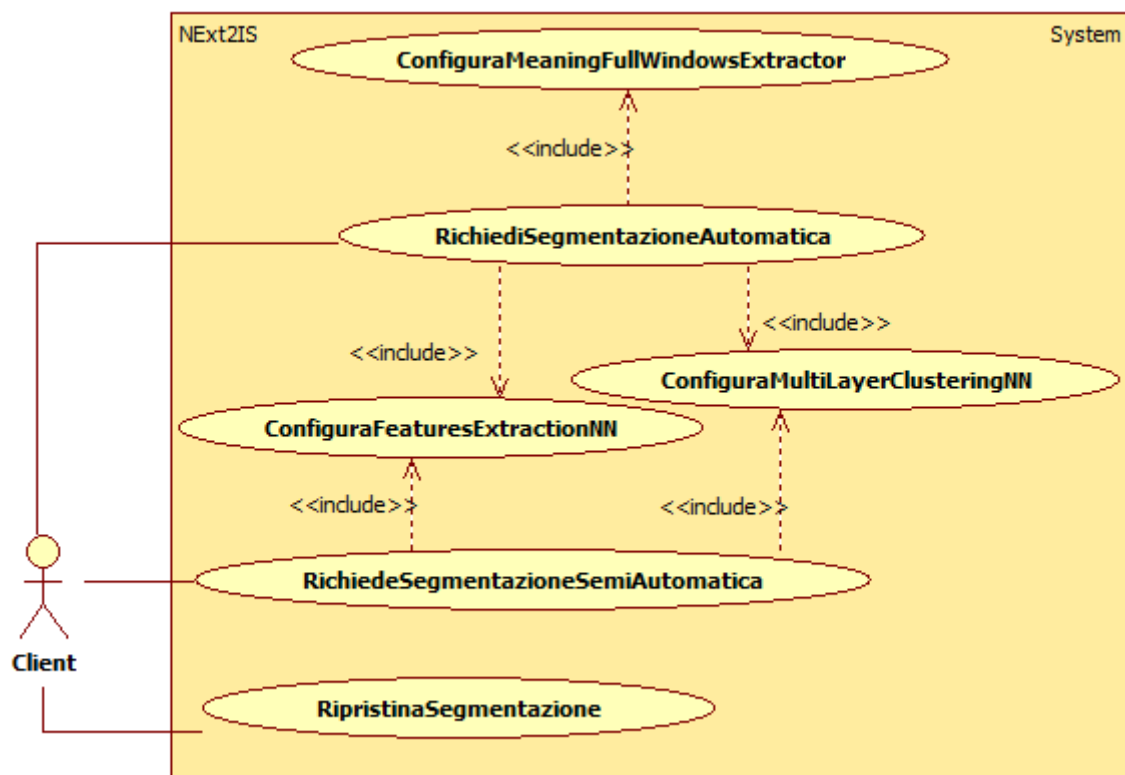


figura 6: Diagramma dei Casi d'uso di NExt2IS

Le descrizioni testuali (secondo il formalismo di Cockburn) dei Casi d'Uso di NExt2IS sono riportate in appendice. Tali Casi d'Uso forniscono una descrizione dettagliata, seppure sintetica, dei requisiti comportamentali richiesti per NExt2IS.

Ad ogni tentativo del Client di inosservanza delle precondizioni dei Casi d'Uso, o di errori nella

specifica dei dati (di configurazione e/o segmentazione), NExt2IS risponderà con il lancio di un'opportuna eccezione.

2.2 Definizione formale di segmentazione

Dal punto di vista astronomico, la segmentazione di un'immagine ci consente di individuare oggetti in un contesto rumoroso. Da un punto di vista matematico, diamo la seguente:

Definizione di segmentazione. Sia F l'insieme di tutti i Pixel che compongono l'immagine da segmentare e sia P un predicato di uniformità (od omogeneità) definito sui gruppi di pixel connessi. Una segmentazione è una partizione di F in una famiglia di sottoinsiemi $(S_1, \dots, S_n)$, tali che $\bigcup S_i = F$, dove:

1. $P(S_i) = \text{vero} \forall i = 1, \dots, n$;
2. $S_i \cap S_j = \emptyset, \forall i, j = 1, \dots, n, i \neq j$;
3. $P(S_i \cup S_j) = \text{falso}, \forall i, j = 1, \dots, n, i \neq j$.

La segmentazione può essere dunque vista come una procedura di *clustering* dei pixel che compongono una immagine. Nel caso di NExt2 per “pixel” si intende la luminosità non solo del pixel in esame, ma anche del contesto in cui si trova: il rumore di cui sono affette le immagini astronomiche è tale per cui un pixel potrebbe risultare scuro ma in un contesto di pixel luminosi (oggetto), o viceversa luminoso in un contesto di pixel scuri (fondo). È evidente che nel primo caso il pixel va correttamente classificato come appartenente a un oggetto, mentre nel secondo caso tale classificazione sarebbe erranea.

2.2.1 Reti Neurali Non Supervisionate

La teoria delle reti neurali è un articolato campo di ricerca nell'ambito dell'intelligenza artificiale: nate come semplice modello delle reti neurali biologiche, le reti neurali artificiali rappresentano un utile e versatile strumento di Data Mining.

In generale le reti neurali sono grafi costituiti da elementi fondamentali detti nodi, o neuroni, che implementano una funzione (in generale, non lineare) dell'input che ricevono. Le connessioni fra i neuroni sono caratterizzate da un numero, il “peso” che definisce la “forza” della connessione, o la distanza fra un neurone e l'altro. I possibili schemi di interconnessione fra due nodi, ovvero le diverse architetture di reti neurali, consentono di affrontare diverse classi di problemi:

- **Classificazione:** attribuzione, a ciascun elemento (pattern) del dataset di un'etichetta di classe. Un esempio di classificazione è la discriminazione di stelle e galassie nei cataloghi astronomici.
- **Regressione:** determinazione della relazione funzionale fra un insieme di variabili indipendenti (spazio dei parametri) e una o più variabili dipendenti (spazio di output).
- **Clustering:** raggruppamento degli elementi di un dataset in classi in modo che gli elementi vicini fra di loro appartengano alla stessa classe.
- **Riduzione della dimensionalità:** per ridurre la complessità del problema in esame può essere utile ridurre la dimensionalità dello spazio dei parametri.

Si definisce “modello” l'insieme di una particolare architettura di rete neurale e dei valori di ciascuna interconnessione (pesi).

2 La segmentazione di un'immagine astronomica

La determinazione del modello è un problema di ottimizzazione che prende il nome di *addestramento*, e che può essere affrontato in svariati modi. Nel seguito definiremo in certo dettaglio solo le procedure di addestramento utili per l'implementazione del segmentatore: tali procedure appartengono ad un'ampia classe di algoritmi di apprendimento *non supervisionato*.

Si definisce *apprendimento supervisionato* un algoritmo di determinazione del modello a partire da un insieme di *esempi* di cui è noto il risultato. L'apprendimento *non supervisionato*, invece, determina, i pesi esclusivamente sulla base delle caratteristiche statistiche dei pattern, senza utilizzare alcuna informazione *a priori*.

L'apprendimento supervisionato è richiesto per risolvere problemi di regressione e classificazione, mentre quello non supervisionato consente di affrontare problemi di clustering e di riduzione della dimensionalità.

Una rete neurale non supervisionata (UnsupervisedNN) ha le seguenti proprietà:

1. conosce la dimensione dello spazio degli input (lunghezza dei Pattern) dei dati in input per risolvere uno specifico problema;
2. ha un numero di neuroni per fornire la soluzione del problema: ad ogni neurone è associato un Pattern, il suo peso, detto Pattern di riferimento;
3. in fase di addestramento si comporta come schematizzato in figura 7.

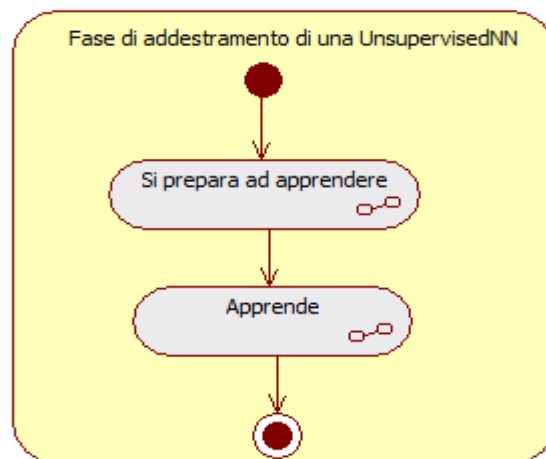


figura 7: Diagramma di Attività per la fase di addestramento di UnsupervisedNN

Le UnsupervisedNN usate per la costruzione del segmentatore (NExt2IS) sono di due tipi (come schematizzato in figura 8):

1. FeaturesExtractionNN, cioè reti neurali non supervisionate per la riduzione di dimensionalità di un Pattern;
2. ClusteringNN, cioè reti neurali non supervisionate per il partizionamento di un'insieme di Pattern.

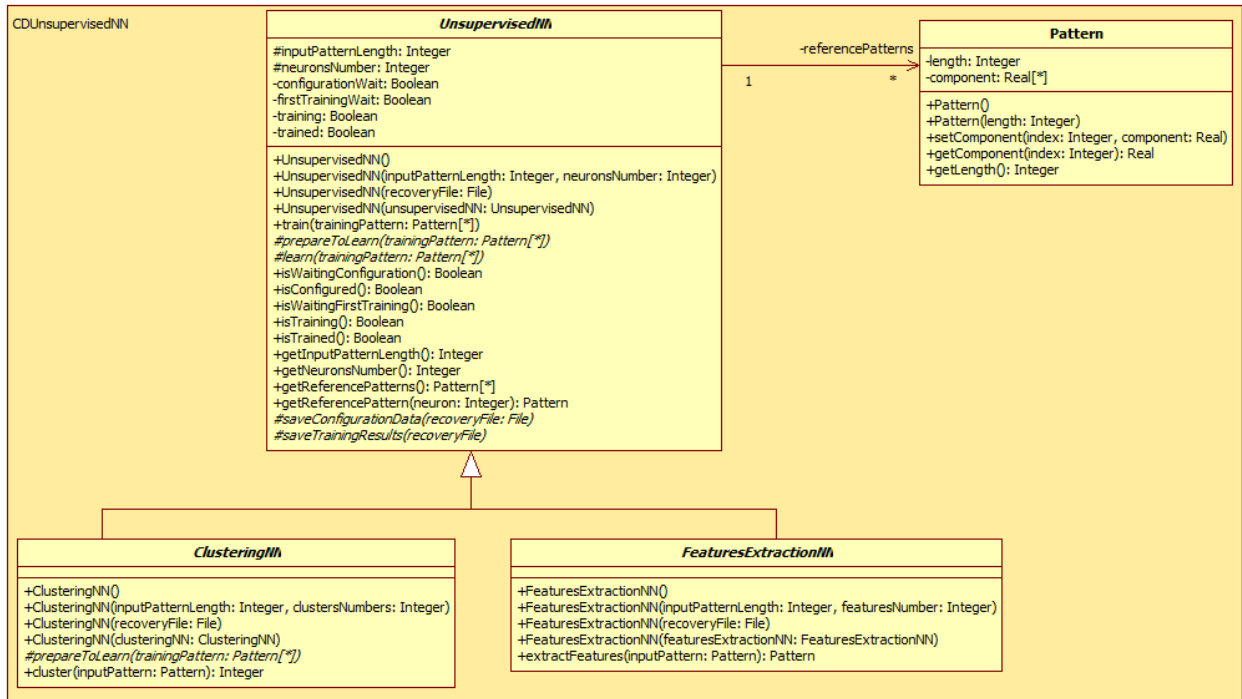


figura 8: Diagramma delle classi delle *UnsupervisedNN*

2.3 Il Clustering dei pixel dell'immagine

Il Clustering dei Pixel è dunque la prima, nonché la più importante, attività per la segmentazione di un'immagine. Il suo input è l'immagine di partenza, il suo output sono i sottoinsiemi (cluster) S_j , della definizione di segmentazione appena data. Per tale attività ci si è avvalso di reti neurali non supervisionate per il clustering di Pattern (sequenze di lunghezza finita, di numeri reali): ogni Pixel dell'immagine dovrà, quindi, essere rappresentato da un Pattern.

2.3.1 Spazio di input e costruzione del dataset

Un Pixel è l'unità elementare dell'informazione contenuta in un'immagine (Image), e corrisponde a un punto luminoso (la relazione tra Image e Pixel è schematizzata in figura 9).

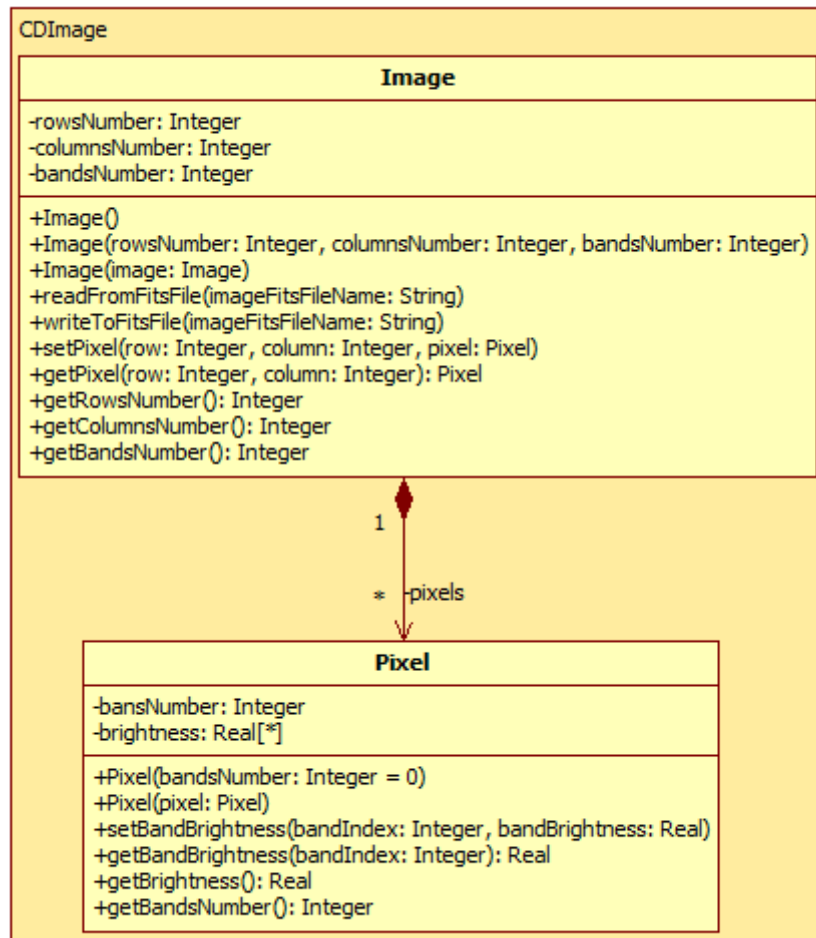


figura 9: Diagramma delle classi che illustra la relazione tra Image

Tale punto luminoso è, a sua volta, il risultato della combinazione di una o più bande di colore. In altre parole, un Pixel è una “sequenza” di una o più luminosità, a seconda del numero di bande che lo costituiscono.

Come accennato in precedenza, quando si segmenta un'immagine si deve tener conto non solo della luminosità del singolo Pixel, ma anche del contesto in cui esso si trova. Più precisamente, per stabilire il cluster di appartenenza di ciascun Pixel, consideriamo anche i Pixel che cadono in un intorno (quadrato) di diametro n , centrato in quel Pixel. Dunque, per rappresentare ciascun Pixel, si fa uso di un Pattern (il cui schema di estrazione è illustrato in figura 10) avente:

- lunghezza uguale a $(n \times n) \times b$, dove
 - (I) n è un numero naturale dispari e maggiore di 1, che rappresenta il diametro dell'intorno del Pixel;
 - (II) b è il numero di bande di ogni Pixel dell'immagine.
- e le cui componenti sono il risultato della concatenazione delle sequenze luminose dei Pixel che cadono nell'intorno (Pixel centrale incluso). L'ordine di concatenazione parte dall'angolo in alto a sinistra dell'intorno e termina nell'angolo in basso a destra dello stesso, procedendo da sinistra a destra e dall'alto verso il basso.

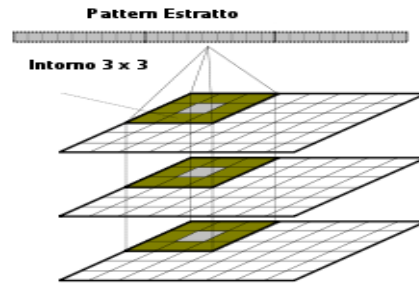


figura 10: Esempio di estrazione di un Pattern da un'immagine su tre bande

Si osservi che, utilizzando tali Pattern, si hanno le seguenti conseguenze principali:

- dalla fase di Clustering, verranno esclusi a priori $(n - 1) / 2$ Pixel per ciascuno dei bordi dell'immagine di input. Tale perdita, però, non influisce sull'esito della segmentazione;
- la lunghezza di ciascun Pattern estratto può essere eccessiva, e rendere poco agevole l'individuazione del cluster di appartenenza. Sorge, pertanto il *problema della riduzione della dimensionalità di un Pattern*.

2.3.2 Riduzione della dimensionalità

Per ridurre la dimensionalità dei Pattern, di volta in volta estratti dall'immagine di input, si è ricorso a reti neurali non supervisionate per l'estrazione delle caratteristiche significative di un Pattern (FeaturesExtractionNN).

L'estrazione delle caratteristiche di un Pattern, è una funzione:

$$f : x \in \mathbb{R}^L \rightarrow y \in \mathbb{R}^M \text{ con } L, M \in \mathbb{N}, L > M$$

tale che:

$$y_j = w_j^T \cdot x,$$

dove x è un Pattern di input, L la sua lunghezza, M il numero di caratteristiche da estrarre, e w_j è il Pattern di riferimento associato al j -esima caratteristica:

$$w_j \in \mathbb{R}^L, \forall j = 1, \dots, M.$$

Reti neurali per l'estrazione delle caratteristiche

Le FeaturesExtractionNN sono UnsupervisedNN che risolvono il problema della riduzione di dimensionalità di un Pattern, la cui principale caratteristica è quella di avere un numero di neuroni inferiore alla lunghezza del Pattern di input.

Un tipo FeaturesExtractionNN (come schematizzato in figura 11) sono le PCANN.

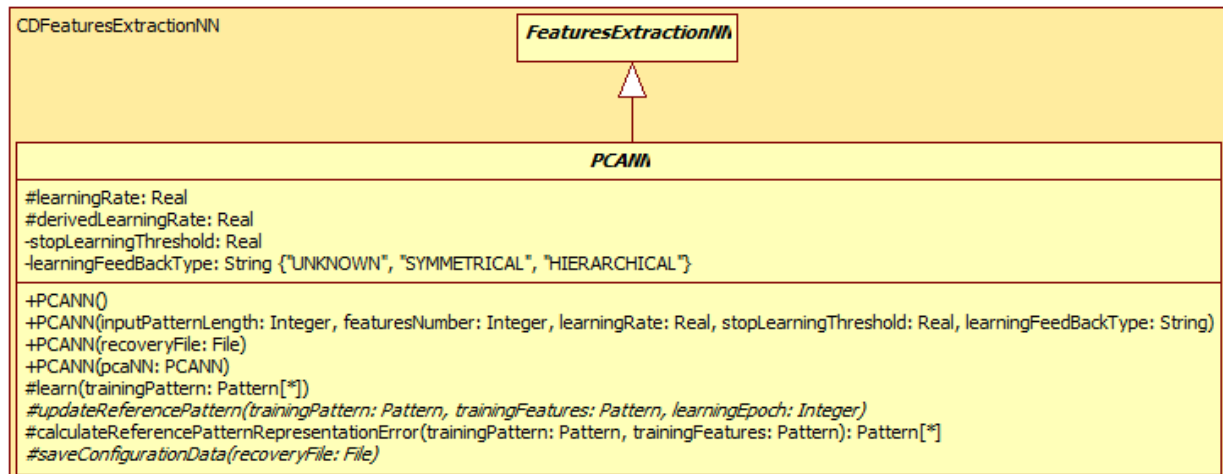


figura 11: Diagramma delle classi della gerarchia delle FeaturesExtractionNN

La PCA

La PCA (Principal Component Analysis) è una tecnica statistica per la riduzione della dimensionalità.

Definizione statistica di PCA. Data una distribuzione statistica di dati in uno spazio L -dimensionale, questa tecnica prende in esame le proprietà della distribuzione e cerca di determinare le componenti che *massimizzano la varianza*, o *minimizzano l'errore di rappresentazione*. Queste componenti vengono dette *componenti principali*, e sono combinazioni lineari di variabili casuali, con la proprietà di massimizzare la varianza in relazione al valore degli *autovalori* (e *autovettori*) della *matrice di covarianza* della distribuzione.

Definizione geometrica di PCA. La PCA corrisponde ad una rotazione degli assi coordinati in un nuovo sistema di coordinate tali che la proiezione dei punti sul primo asse abbia la prima varianza massima, la proiezione sul secondo asse abbia la seconda varianza massima e così via (si veda la figura 12).

Definizione matematica di PCA. Sia $x \in \mathbb{R}^L$ ottenuto da una qualche distribuzione centrata nella media $E(x)=0$ e sia $C=E(xx^T)$ la *matrice di covarianza* della distribuzione. L' i -esima *componente principale* di x è definita come $\phi_i^T x$, dove ϕ_i rappresenta l'autovettore normalizzato di C corrispondente all' i -esimo più grande autovalore λ_i . Il sottospazio ottenuto dagli autovettori $\phi_1, \dots, \phi_M$ con $M < L$, è chiamato *sottospazio PCA di dimensione M*.

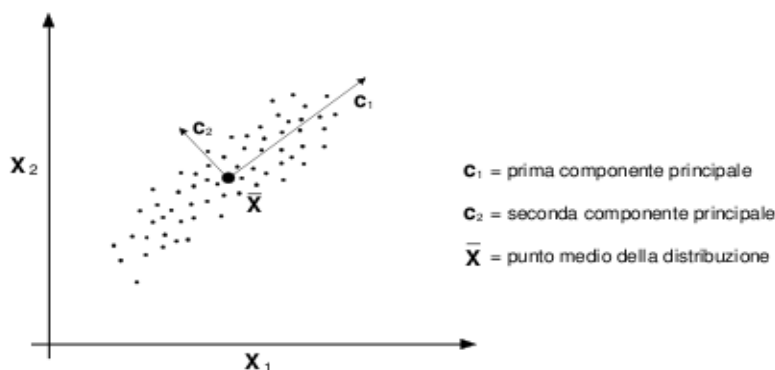


figura 12: Esempio di PCA su Pattern bidimensionali

Reti neurali PCA

La PCA può essere realizzata tramite una rete neurale, in cui i Pattern di riferimento associati ai neuroni convergono, in fase di apprendimento, agli autovettori principali ϕ_j per $j=1, \dots, M$. Tali reti hanno un apprendimento di tipo *hebbiano* (di rinforzo), cioè il Pattern di riferimento, corrispondente a un neurone, è aggiornato all'attivazione dell'input e dell'output di quel neurone.

Una rete neurale PCA (PCANN) è una FeaturesExtractionNN con le seguenti proprietà:

1. un *tasso di apprendimento*, μ_{ω} , che può assumere un valore reale compreso tra 0 e 1, estremi esclusi;
2. una *soglia per l'arresto (convergenza) della fase di apprendimento* ξ (detta anche *soglia del livello di apprendimento* η), che può assumere un valore reale compreso tra 0 e 1, estremi esclusi;
3. il tipo di feedback da usare durante la fase di apprendimento (illustrato in figura 13), che può essere :
 - (I) gerarchico: ogni neurone invia il suo output a se stesso ed ai neuroni con indici maggiori;
 - (II) simmetrico: ogni neurone invia il suo output a se stesso ed a tutti gli altri neuroni;
4. livello di apprendimento η è dato della seguente formula:

$$\eta = \sqrt{\sum_{j=1}^M \|w_j^{(epoch)} - w_j^{(epoch-1)}\|_2^2}, \quad epoch > 0$$

5. la complessità di apprendimento, espressa in numero di operazioni floating-point, è pari a

$$O(|TS| \cdot Epochs_{\xi} \cdot M \cdot L)$$

dove

- TS è la sequenza dei Pattern di addestramento (Training Set);
- $Epochs_{\xi}$ è il numero di epoche (iterazioni) affinché $\eta \leq \xi$.

La fase di apprendimento per una generica PCANN è schematizzata in figura 14. Si osservi che l'attività "Aggiorna i Pattern di riferimento", evidenziata in grigio in figura 14 dipende dallo specifico tipo di PCANN che si ha intenzione di usare. Le PCANN si distinguono fondamentalmente in due tipi (come schematizzato in figura 15):

1. PCANN Standard;
2. PCANN Non Standard.

2 La segmentazione di un'immagine astronomica

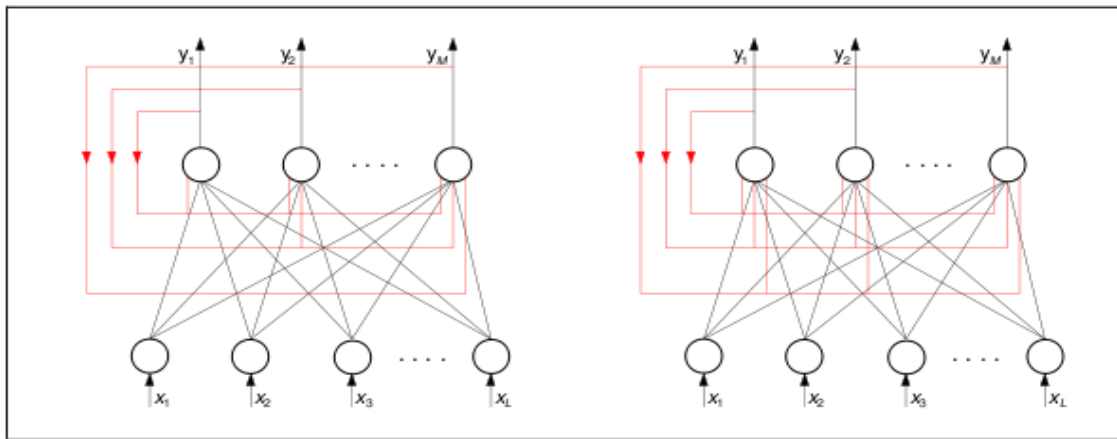


figura 13: PCANN gerarchica, a sinistra; PCANN simmetrica, a destra

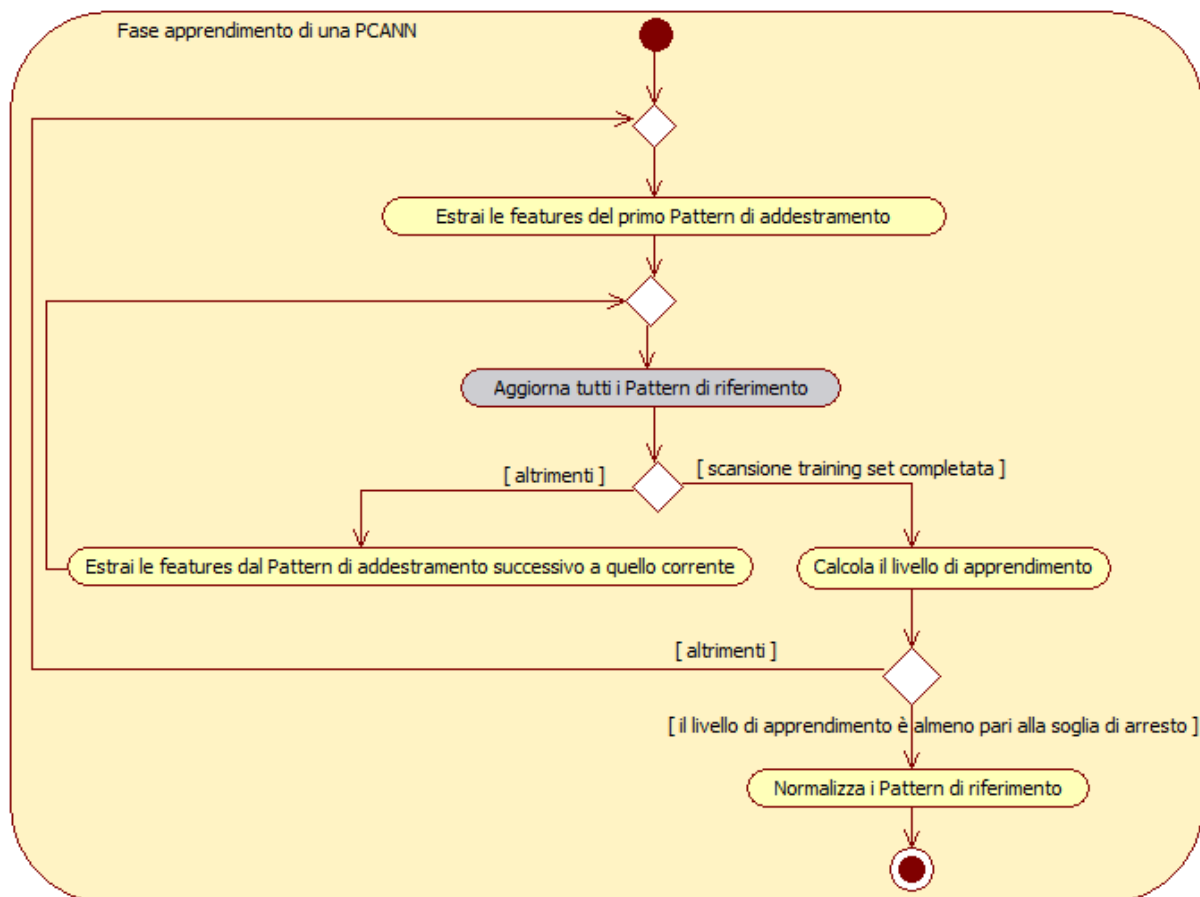


figura 14: Diagramma delle Attività di una PCANN

2 La segmentazione di un'immagine astronomica

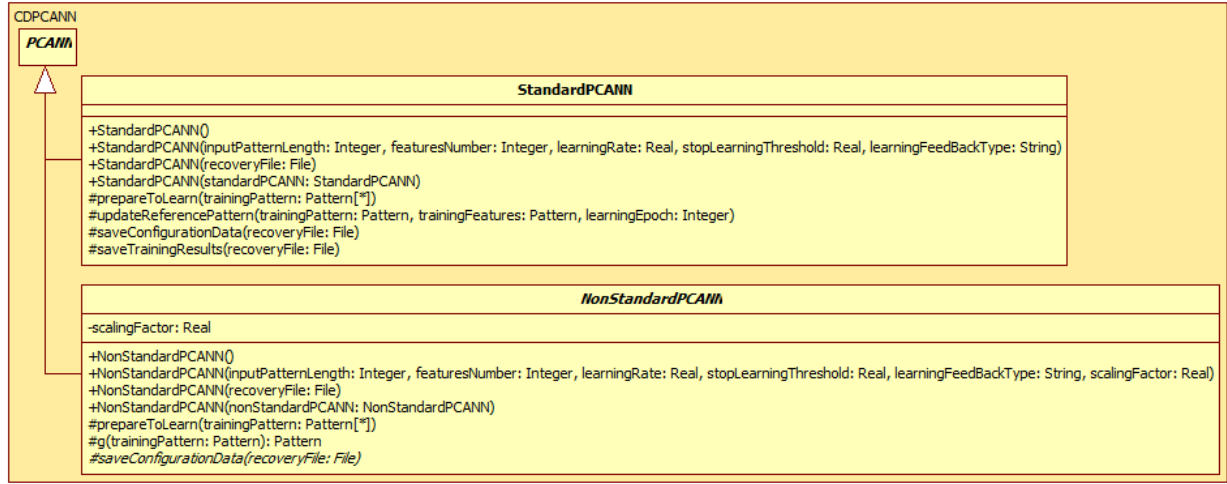


figura 15: Diagramma delle classi della gerarchia delle PCANN

PCANN Standard

Le PCANN Standard sono PCANN in cui Pattern di riferimento sono aggiornati in base alla formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \mu^{(epoch)} \cdot y_j^{((t-1)+epoch)} \cdot e_j^{((t-1)+epoch)}, \quad \forall j \in [1..M], \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \text{numero casuale}, \quad \forall j \in [1..M]$$

dove:

1. w_j è il Pattern di riferimento corrispondente al j -iesimo neurone;
2. t è l'indice, nella sequenza di addestramento (TS), del Pattern di addestramento utilizzato per l'aggiornamento;
3. $epoch$ è l'indice della corrente iterazione di apprendimento;
4. μ è il tasso di apprendimento derivato, la cui formula è la seguente:

$$\mu^{(0)} = \mu_w \cdot \max_{p \in TS} \left| \sum_{i=1}^L p_i \right|;$$

$$\mu^{(epoch)} = 0.9 \cdot \mu^{(epoch-1)}, \quad \forall epoch > 0$$

5. y è il Pattern contenente le caratteristiche estratte dal t -iesimo Pattern di TS;
6. e_j è una stima dell'errore istantaneo di rappresentazione per la j -iesima caratteristica, la cui formula è la seguente:

$$e_j^{((t-1)+epoch)} = x^{((t-1)+epoch)} - \sum_{k=1}^{l(j)} y_k^{((t-1)+epoch)} \cdot w_k^{((t-1)+epoch)}$$

dove:

- $l(j) = j$, nel caso di PCANN Standard con feedback di apprendimento di tipo gerarchico;
- $l(j) = M$, nel caso di PCANN Standard con feedback di apprendimento di tipo simmetrico.

PCANN Non Standard

Le PCANN Non Standard, sono PCANN nate per superare alcune limitazioni delle PCANN Standard. Queste ultime, infatti:

1. sono abili a realizzare solo corrispondenze input-output lineari;
2. non sono in grado di separare sotto-segnali indipendenti dalle loro combinazioni reali.

Inoltre, le PCANN Non Standard si distinguono da quelle Standard, per le seguenti proprietà:

1. una funzione $g(z)$, dispari e non decrescente in z . Tale funzione è la derivata di una funzione costo $f(z)$, pari, non negativa, quasi ovunque continua, differenziabile e che cresca più lentamente di z^2 . Nel nostro caso si è scelto:

$$g(z) = \tanh(\alpha \cdot z)$$

dove, α è un fattore di scala che assume valori compresi tra 0 e 1, estremi inclusi. Tale funzione viene usata durante la fase di apprendimento, e il dato z su cui opera dipende dal tipo di PCANN Non Standard usata;

2. il tasso di apprendimento derivato μ , per l'aggiornamento dei Pattern di riferimento, è dato dalla formula:

$$\mu^{(0)} = \mu_w \cdot \max_{p \in TS} \left| \sum_{i=1}^L g(p_i) \right|;$$

$$\mu^{(epoch)} = 0.9 \cdot \mu^{(epoch-1)}, \quad \forall epoch > 0$$

3. si suddividono in due categorie (come schematizzato in figura 16):

- (I) PCANN Robusta;
- (II) PCANN Non Lineare.

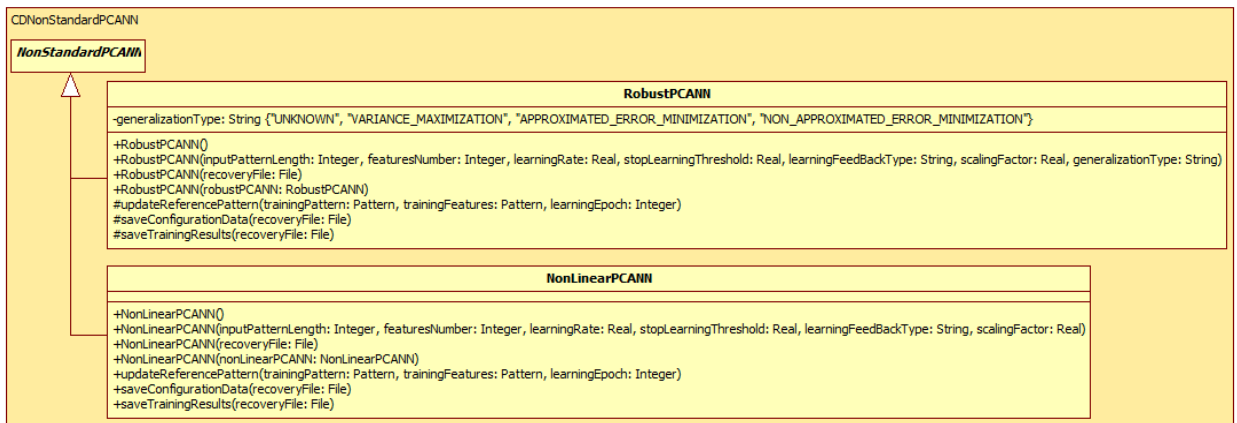


figura 16: Diagramma delle classi della gerarchia delle NonStandardPCANN

PCANN Robusta

Le PCANN Robuste, sono PCANN Non Standard con le seguenti proprietà:

1. la funzione g viene applicata, o meno, all'output di ciascun neurone a seconda del tipo di generalizzazione scelto per la fase di apprendimento;
2. il tipo di generalizzazione per la fase di apprendimento può essere:

(I) *massimizzazione della varianza*, e in tal caso i Pattern di riferimento sono aggiornati in base alla formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \mu^{(epoch)} \cdot g(y_j^{((t-1)+epoch)}) \cdot e_j^{((t-1)+epoch)}, \quad \forall j \in [1..M], \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \text{numero casuale}, \quad \forall j \in [1..M]$$

(II) *minimizzazione dell'errore non approssimato*, e in tal caso i Pattern di riferimento sono aggiornati in base alla formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \mu^{(epoch)} \cdot (w_j^{((t-1)+epoch)T} \cdot g(e_j^{((t-1)+epoch)}) \cdot x^{(t+epoch)} + x^{(t+epoch)T} \cdot w_j^{((t-1)+epoch)} \cdot g(e_j^{((t-1)+epoch)})), \quad \forall j \in [1..M], \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \text{numero casuale}, \quad \forall j \in [1..M]$$

(III) *minimizzazione dell'errore approssimato*, e in tal caso i Pattern di riferimento sono aggiornati in base alla formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \mu^{(epoch)} \cdot y_j^{((t-1)+epoch)} \cdot g(e_j^{((t-1)+epoch)}), \quad \forall j \in [1..M], \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \text{numero casuale}, \quad \forall j \in [1..M]$$

dove tutti i termini usati nelle formule sopracitate hanno la medesima definizione di quelli usati nella formula di aggiornamento dei Pattern di riferimento, relativa ad una PCANN Standard.

PCANN Non Lineari

Le PCANN Non Lineari sono PCANN Non Standard i cui Pattern di riferimento sono aggiornati, in fase di apprendimento, secondo la formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \mu \cdot g(y_j^{((t-1)+epoch)}) \cdot e_j^{((t-1)+epoch)}, \quad \forall j \in [1..M], \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \text{numero casuale}, \quad \forall j \in [1..M]$$

dove tutti i termini usati nelle formule sopracitate hanno la medesima definizione di quelli usati nella formula di aggiornamento dei Pattern di riferimento, relativa ad una PCANN Standard, ad eccezione della formula per il calcolo di e_j , che viene sostituita da:

$$e_j^{((t-1)+epoch)} = x^{((t-1)+epoch)} - \sum_{k=1}^{l(j)} g(y_k^{((t-1)+epoch)}) \cdot w_k^{((t-1)+epoch)}$$

2.3.3 Reti neurali non supervisionate per il Clustering

Il clustering è, in genere, definito come la suddivisione non supervisionata di un insieme di punti nello spazio degli input in “classi di vicinanza”. In NExt2 questo compito viene svolto per mezzo

di reti neurali non supervisionate.

Le ClusteringNN sono UnsupervisedNN che risolvono il problema del partizionamento di un insieme di Pattern, la cui principale caratteristica è quella di avere un numero di neuroni non inferiore a due, indipendentemente dalla lunghezza specificata per i Pattern di input.

L'individuazione del cluster di appartenenza di un Pattern x , avviene mediante l'applicazione della formula:

$$x \in S_k \Leftrightarrow k \in [1..C] \text{ e } \|x - w_k\|_2^2 \leq \|x - w_j\|_2^2 \forall j \in [1.., C].$$

dove C è il numero di neuroni (cluster) della rete, w_j è, il Pattern di riferimento associato al j -esimo neurone.

Le ClusteringNN si distinguono fondamentalmente in due tipi (come schematizzato in figura 17):

1. FrozenMorphologyClusteringNN: reti neurali non supervisionate per il clustering la cui morfologia non cambia tra un addestramento e il successivo;
2. ChangeableMorphologyClusteringNN: reti neurali non supervisionate per il clustering la cui morfologia cambia tra un addestramento e il successivo.

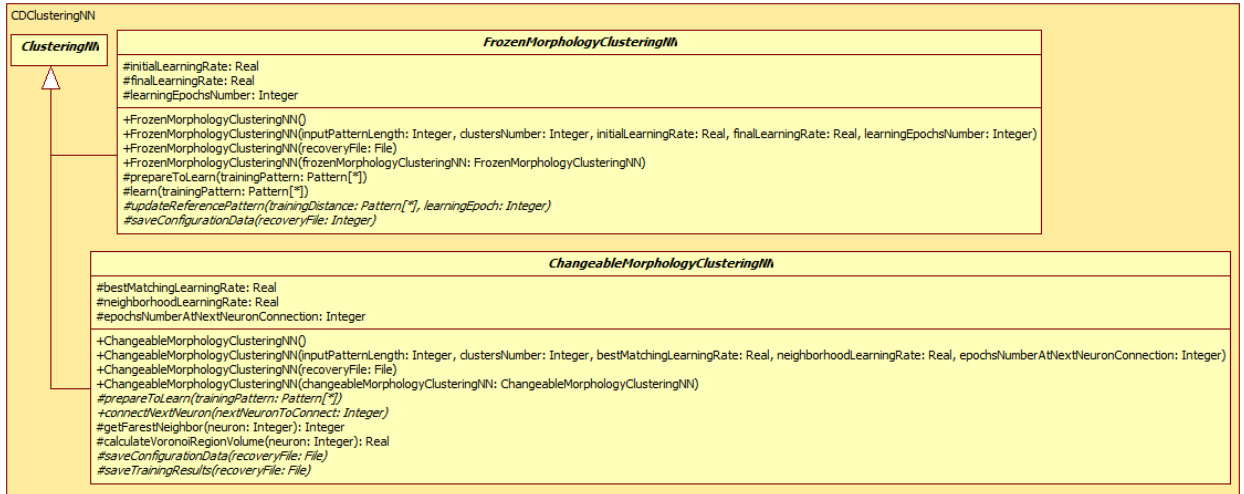


figura 17: Diagramma delle classi per la gerarchia delle ClusteringNN

Reti neurali non supervisionate per il clustering a morfologia fissa

Le FrozenMorphologyClusteringNN sono ClusteringNN, la cui morfologia non cambia durante la fase di addestramento. Una FrozenMorphologyClusteringNN ha le seguenti proprietà:

- un tasso di apprendimento iniziale ε_i , rappresentato da un numero reale con un valore compreso tra 0 e 1, estremi esclusi;
- un tasso di apprendimento finale ε_f , rappresentato da un numero reale con un valore almeno pari a 0, ma inferiore a 1, e minore, o al più uguale, a ε_i ;
- il numero di epoche $Epoch_{max}$, necessarie per completare l'apprendimento;
- il comportamento, in fase di apprendimento, schematizzato in figura 18.

Le FrozenMorphologyClusteringNN sono di quattro tipi (come schematizzato in figura 19):

- KMeansClusteringNN;
- MaximumEntropyClusteringNN;

2 La segmentazione di un'immagine astronomica

- NeuralGasClusteringNN;
- SOMClusteringNN.

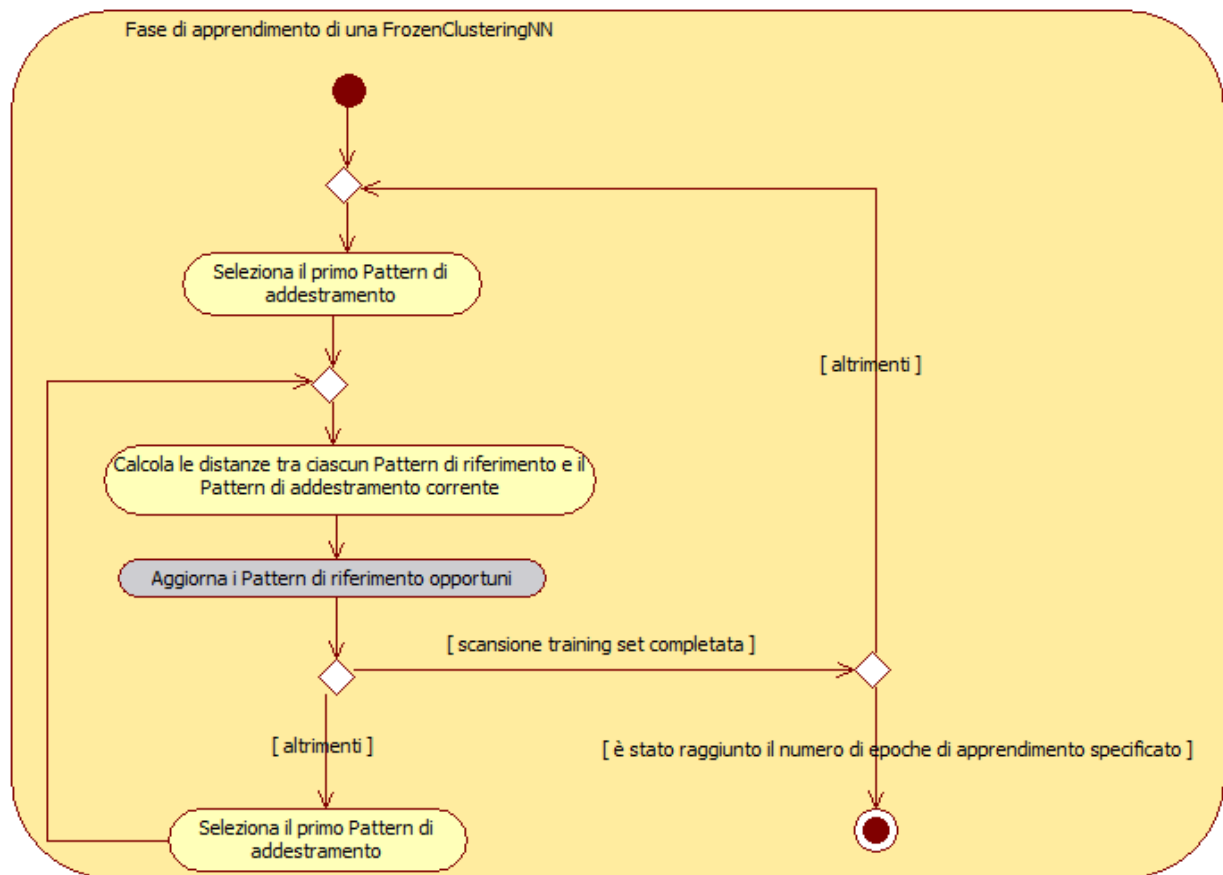
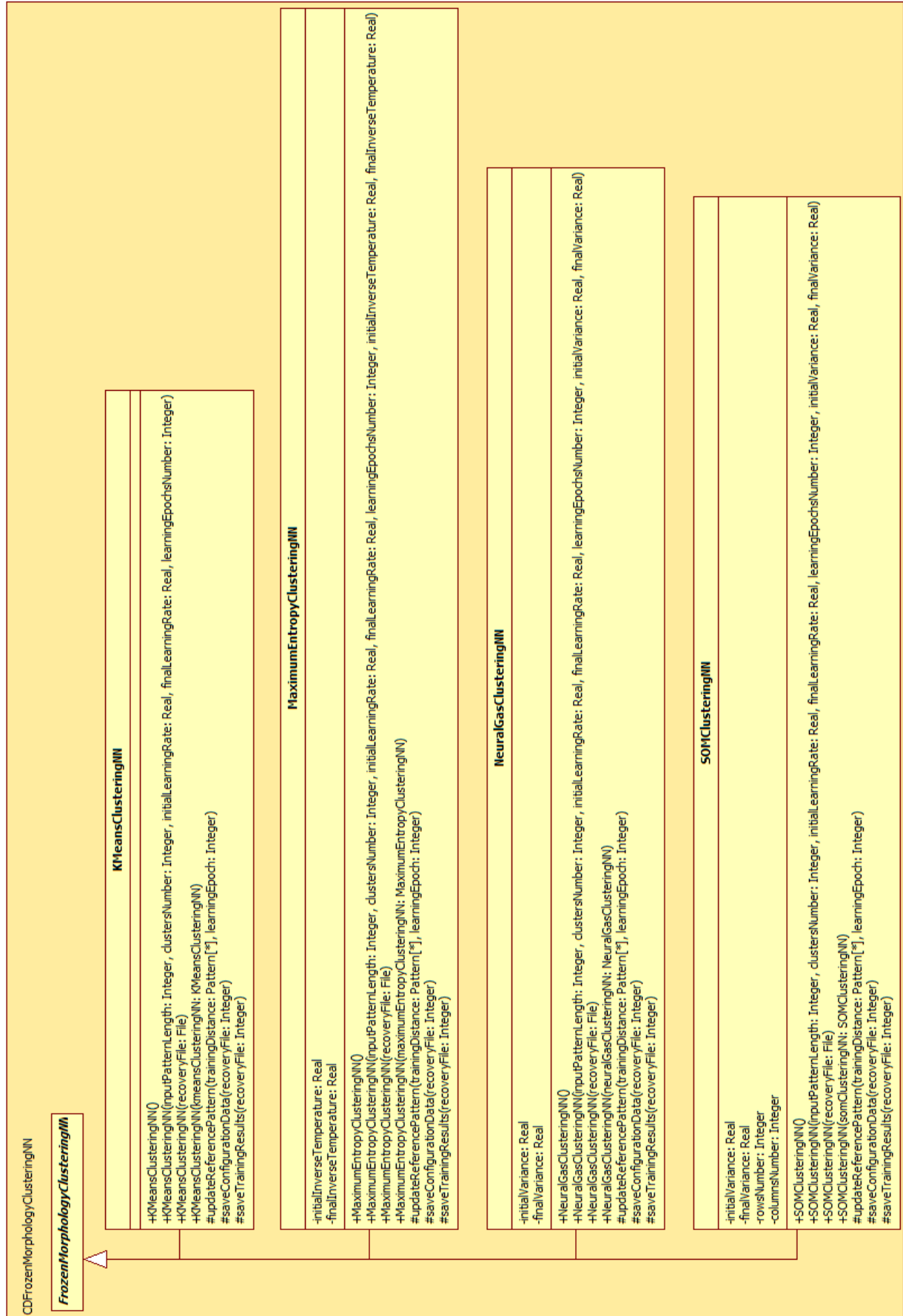


figura 18: Diagramma delle Attività della fase di addestramento per una FrozenClusteringNN

figura 19: Diagramma delle classi della gerarchia delle **FrozenMorphologyClusteringNN**

Reti neurali KMeans

Le KMeansClusteringNN sono FrozenMorphologyClusteringNN i cui Pattern di riferimento sono aggiornati in base alla formula:

$$w_k^{(t+epoch)} = w_k^{((t-1)+epoch)} + \epsilon^{(epoch)} \cdot (x^{(t)} - w_k^{((t-1)+epoch)}), \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \left(\frac{\nu}{\nu_{max}}\right) \cdot (\gamma_{max} - \gamma_{min}) + \gamma_{min}, \quad \forall j \in [1..C]$$

dove

- con C che indica il numero di neuroni della rete,

$$k = \underset{j \in [1..C]}{\operatorname{argmin}} \|x^{(t)} - w_j^{((t-1)+epoch)}\|_2$$

- ν indica un numero casuale;
- ν_{max} il massimo numero casuale generabile;
- con TS che indica la sequenza dei Pattern di addestramento (Training Set), e M indicante la lunghezza del Pattern di input:

$$\gamma_{min} = \min_{i \in [1..M]} p_i, \quad \forall p \in TS;$$

$$\gamma_{max} = \max_{i \in [1..M]} p_i, \quad \forall p \in TS.$$

- il *tasso di apprendimento corrente* $\epsilon^{(epoch)}$ è pari a:

$$\epsilon^{(epoch)} = \left(\epsilon_i \cdot \left(\frac{\epsilon_f}{\epsilon_i}\right)^{\left(\frac{epoch}{Epoch_{max}}\right)}\right)$$

- t è l'indice, nella sequenza di addestramento (TS), del Pattern di addestramento utilizzato per l'aggiornamento.

Le KmeansClusteringNN hanno una complessità di apprendimento, espressa in numero di operazioni floating-point, pari a:

$$O(|TS| \cdot Epoch_{max} \cdot C \cdot M)$$

Reti Neurali Maximum Entropy

Le MaximumEntropyClusteringNN sono FrozenMorphologyClusteringNN con le seguenti proprietà:

- una *temperatura inversa iniziale* β_i per la fase di apprendimento, con valore reale positivo;
- una *temperatura inversa finale* β_f per la fase di apprendimento, con valore reale positivo superiore, o al più pari, a β_i ;
- l'aggiornamento dei Pattern di riferimento avviene in base alla formula:

2 La segmentazione di un'immagine astronomica

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \epsilon^{(epoch)} \cdot E_j^{(t+epoch)} \cdot (x^{(t)} - w_j^{((t-1)+epoch)}), \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \left(\frac{\nu}{\nu_{max}}\right) \cdot (\gamma_{max} - \gamma_{min}) + \gamma_{min}, \quad \forall j \in [1..C]$$

dove:

- l'entropia corrente $E_j^{(t+epoch)}$ del j -esimo neurone definita come:

$$E_j^{(t+epoch)} = \frac{e^{(-\beta^{(epoch)} \cdot d_j)}}{\sum_{k=1}^C e^{(-\beta^{(epoch)} \cdot d_k^{(t+epoch)})}};$$

$$d_j^{(t+epoch)} = \frac{\|x^{(t)} - w_j^{((t-1)+epoch)}\|_2^2}{\max_{h \in [1..C]} \|x^{(t)} - w_h^{((t-1)+epoch)}\|_2^2}$$

- la temperatura inversa corrente $\beta^{(epoch)}$ pari a:

$$\beta^{(epoch)} = \left(\beta_i \cdot \left(\frac{\beta_f}{\beta_i}\right)^{\frac{epoch}{Epoch_{max}}}\right)$$

- tutti gli altri termini definiti come per le reti KMeans

Le MaximumEntropyClusteringNN hanno una complessità di apprendimento, espressa in numero di operazioni floating-point, pari a:

$$O(|TS| \cdot Epoch_{max} \cdot C \cdot M)$$

Reti Neurali Neural Gas

Le NeuralGasClusteringNN sono FrozenMorphologyClusteringNN con le seguenti proprietà:

- una *varianza iniziale* σ_i per la fase di apprendimento, con valore reale positivo;
- una *varianza finale* σ_f per la fase di apprendimento, con valore reale positivo inferiore, o al più pari, a σ_i ;
- l'aggiornamento dei Pattern di riferimento avviene in base alla formula:

$$w_j^{(t+epoch)} = w_j^{((t-1)+epoch)} + \epsilon^{(epoch)} \cdot h_{\sigma^{(epoch)}}(rank(j)) \cdot (x^{(t)} - w_j^{((t-1)+epoch)}), \quad (t+epoch) > 0;$$

$$w_j^{(0)} = \left(\frac{\nu}{\nu_{max}}\right) \cdot (\gamma_{max} - \gamma_{min}) + \gamma_{min}, \quad \forall j \in [1..C]$$

dove:

- h_{σ} è la funzione unimodale decrescente, con varianza $\sigma^{(epoch)}$:

$$h_{\sigma^{(epoch)}}(rank(j)) = e^{\left(\frac{-rank(j)}{\sigma^{(epoch)}}\right)}$$

- siano

- $D = \langle d_1, \dots, d_C \rangle$ tale che $d_j = (x^{(t)} - w_j^{((t-1)+epoch)}) \quad \forall j \in [1..C]$;

- $D' = \langle d_{k1}', \dots, d_{kC}' \rangle$ tale che

2 La segmentazione di un'immagine astronomica

- $\forall h \in [1..C] \exists! j \in [1..C]$ per cui $k_h = j$;
- $\forall h \in [1..(C - 1)] \|d_{kh}\|_2^2 \leq \|d_{k(h+1)}\|_2^2$;

allora $rank(j) = h \in [1..C] \mid k_h = j$.

- la *varianza corrente* $\sigma^{(epoch)}$ pari a:

$$\sigma^{(epoch)} = \left(\sigma_i \cdot \left(\frac{\sigma_f}{\sigma_i} \right)^{\left(\frac{epoch}{Epoch_{max}} \right)} \right).$$

- tutti gli altri termini definiti come per le reti KMeans.

Le NeuralGasClusteringNN hanno una complessità di apprendimento, espressa in numero di operazioni floating-point, pari a:

$$O(|TS| \cdot Epoch_{max} \cdot Clog(C) \cdot M)$$

Reti neurali SOM

Le SOMClusteringNN (Self Organizing Map ClusteringNN) sono FrozenMorphologyClusteringNN con le seguenti proprietà:

- una *varianza iniziale* σ_i per la fase di apprendimento, con valore reale positivo;
- una *varianza finale* σ_f per la fase di apprendimento, con valore reale positivo inferiore, o al più pari, a σ_i ;
- i neuroni (e, di conseguenza, i Pattern di riferimento a loro) organizzati in una griglia di P righe e Q colonne, tali che $P \times Q =$ numero di neuroni specificato per la rete;
- l'aggiornamento dei Pattern di riferimento avviene in base alla formula:

$$w_{(i,j)}^{(t+epoch)} = w_{(i,j)}^{((t-1)+epoch)} + \epsilon^{(epoch)} \cdot h_{\sigma^{(epoch)}}(|i-k|^2 + |j-h|^2) \cdot (x^{(t)} - w_{(i,j)}^{((t-1)+epoch)}), \quad (t+epoch) > 0;$$

$$w_{(i,j)}^{(0)} = \left(\frac{V}{V_{max}} \right) \cdot (Y_{max} - Y_{min}) + Y_{min}, \quad \forall i \in [1..P] \text{ e } j \in [1..Q]$$

dove:

- h_{σ} è la funzione unimodale decrescente, con varianza $\sigma^{(epoch)}$:

$$h_{\sigma^{(epoch)}}(|i-k|^2 + |j-h|^2) = e^{\frac{-(|i-k|^2 + |j-h|^2)}{2(\sigma^{(epoch)})^2}}$$

- $k \in [1..P]$ e $h \in [1..Q]$, e sono tali che

$$\|x^{(t)} - w_{(k,h)}^{((t-1)+epoch)}\|_2^2 \leq \|x^{(t)} - w_{(i,j)}^{((t-1)+epoch)}\|_2^2 \quad \forall i \in [1..P] \text{ e } j \in [1..Q]$$

- la *varianza corrente* $\sigma^{(epoch)}$ è definita come per le reti Neural Gas;
- tutti gli altri termini definiti come per le reti KMeans.

Le SOMClusteringNN hanno una complessità di apprendimento, espressa in numero di operazioni

floating-point, pari a:

$$O(|TS| \cdot Epoch_{max} \cdot C \cdot M)$$

Reti neurali non supervisionate per il clustering a morfologia variabile

Le ChangeableMorphologyClusteringNN sono ClusteringNN, la cui morfologia cambia durante la fase di addestramento. Al termine dell'addestramento la rete risulta rappresentata da un grafo $G = (V, E)$ tale che

- $V = \{1, \dots, M\}$ è l'insieme degli M neuroni specificati per la rete;
- $E =$ insieme delle connessioni tra i neuroni della rete;
- G non orientato e connesso.

Una ChangeableMorphologyClusteringNN ha le seguenti proprietà:

- un *tasso di apprendimento best-matching* ϵ_b , rappresentato da un numero reale con un valore compreso tra 0 e 1, estremi esclusi. Esso viene usato per aggiornare il Pattern di riferimento corrispondente al cluster k (neurone best-matching) di appartenenza del Pattern di addestramento corrente:

$$k \in [1..C_{connected}] \text{ e } \|x^{(epoch \bmod |TS|+1)} - w_k^{(C_{connected}+epoch)}\|_2^2 \leq \|x^{(epoch \bmod |TS|+1)} - w_j^{(C_{connected}+epoch)}\|_2^2, \forall j \in [1..C_{connected}].$$

dove $C_{connected}$ è il numero di neuroni connessi al momento dell'aggiornamento, TS la sequenza dei Pattern di addestramento (Training Set);

- un *tasso di apprendimento neighborhood* ϵ_n , rappresentato da un numero reale con un valore almeno pari a 0, ma inferiore a 1, e minore, o al più uguale, a ϵ_b . Esso viene usato per aggiornare i Pattern di riferimento corrispondenti ai neuroni direttamente connessi al best-matching (cioè i suoi vicini);
- il numero di epoche di apprendimento $Epoch_{connection}$, necessarie per aggiornare i Pattern di riferimento (best-matching e neighborhood) prima di connettere il prossimo neurone ad almeno uno di quelli già connessi;
- il numero di neuroni C deve essere almeno pari ad un minimo C_{min} , che dipende dal tipo ChangeableMorphologyClusteringNN adottato;
- l'aggiornamento dei Pattern di riferimento avviene, ad ogni epoca di apprendimento, in base alla formula:

$$w_k^{(C_{connected}+epoch)} = w_k^{(C_{connected}+epoch)} + \epsilon_b \cdot (x^{(epoch \bmod |TS|+1)} - w_k^{(C_{connected}+epoch)});$$

$$w_{n_k} = w_{n_k} + \epsilon_n \cdot (x^{(epoch \bmod |TS|+1)} - w_{n_k}^{(C_{connected}+epoch)}), \in [1..C] \forall n_k \in N_k.$$

dove N_k è la regione di Voronoi del neurone best-matching, cioè l'insieme di tutti i suoi vicini;

- ciascun neurone dopo il C_{min} -iesimo viene connesso dopo $Epoch_{connection}$ aggiornamenti dei Pattern di riferimento.

Un tipo ChangeableMorphologyClusteringNN (come schematizzato in figura 20) sono le GCSClusteringNN.

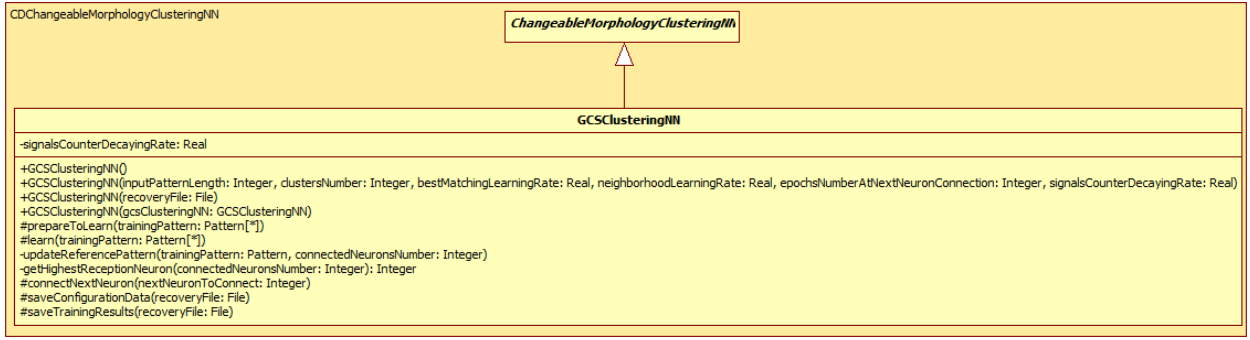


figura 20: Diagramma delle classi della gerarchia delle ChangeableMorphologyClusteringNN

Reti neurali GCS

Le GCSClusteringNN (Growing Cell Structure ClusteringNN) sono ChangeableMorphologyClusteringNN, basate sull'uso del *simplezzo bidimensionale*, con le seguenti proprietà:

- ogni modifica alla rete (aggiunta o rimozione di nodi) mantiene sempre intatte le proprietà del *simplezzo bidimensionale*, ossia:

$$\forall j \in V \quad \exists j_1, j_2 \in V - \{j\}, \quad j_1 \neq j_2 \quad \text{tale che} \quad (j, j_1), (j, j_2), (j_1, j_2) \in E;$$

- il *tasso di decadimento* α_τ del contatore di segnali ricevuti τ_j , associato a ciascun neurone ed utilizzati solo in fase di apprendimento;
- il *numero di neuroni* C deve essere almeno pari a $C_{min} = 3$;
- durante l'aggiornamento dei Pattern di riferimento, il contatore dei segnali ricevuti, associato a ciascun neurone è aggiornato mediante la formula:

$$\begin{aligned} \tau_k^{(C_{connected} + epoch)} &= (1 + \alpha_\tau) \tau_k^{(C_{connected} + (epoch - 1))}; \\ \tau_j^{(C_{connected} + epoch)} &= 1 - \alpha_\tau \tau_j^{(C_{connected} + (epoch - 1))}, \quad \forall j \in [1..C_{connected}] - \{k\} \end{aligned}$$

- la connessione di un neurone avviene come segue:
 - sia $C_{connected} < C$ e $h \in [1..C_{connected}]$ tale che $\tau_h \geq \tau_j \forall j \in [1..C_{connected}]$;
 - sia $f_h \in N_h$ tale che $\forall n_h \in N_h - \{f_h\}$

$$k \in [1..C_{connected}] \quad e \quad \|w_h^{(C_{connected} + Epoch_{connection})} - w_{f_h}^{(C_{connected} + Epoch_{connection})}\|_2^2 \leq \|w_h^{(C_{connected} + Epoch_{connection})} - w_{n_h}^{(C_{connected} + Epoch_{connection})}\|_2^2;$$

allora, detti V_j^{old} e V_j^{new} , il volume della regione di Voronoi del j -esimo neurone ($j \in [1..C_{connected}]$), rispettivamente, prima e dopo la connessione del $(C_{connected} + 1)$ -esimo neurone, si ha che:

- $w_{(C_{connected} + 1)}^{(C_{connected} + Epoch_{connection})} = \frac{1}{2} (w_h^{(C_{connected} + Epoch_{connection})} + w_{f_h}^{(C_{connected} + Epoch_{connection})});$
- $V_{Voronoi_j}^{old} = \left(\frac{1}{|N_j|} \cdot \sum_{n_j \in N_j} \|w_j^{(C_{connected} + Epoch_{connection})} - w_{n_j}^{(C_{connected} + Epoch_{connection})}\|_2^2 \right)^{\frac{M}{2}}, \quad \forall j \in [1..C_{connected}];$

2 La segmentazione di un'immagine astronomica

- $N_h = N_h - f_h, \quad N_{f_h} = N_{f_h} - h, \quad N_{(C_{connected}+1)} = (N_{f_h} \cap N_h) \cup f_h \cup h;$
- $V_{Voronoi_j}^{new} = \left(\frac{1}{|N_j|} \cdot \sum_{n_j \in N_j} \|w_j^{(C_{connected}+Epoch_{connection})} - w_{n_j}^{(C_{connected}+Epoch_{connection})}\|_2 \right)^M, \quad \forall j \in [1..C_{connected}];$
 $\tau_j^{(C_{connected}+1)} = \tau_j^{(C_{connected}+Epoch_{connection})} + \Delta \tau_j^{(C_{connected}+1)};$
- $\Delta \tau_j^{(C_{connected}+1)} = \tau_j^{(C_{connected}+Epoch_{connection})} \left(\frac{V_{Voronoi_j}^{new} - V_{Voronoi_j}^{old}}{V_{Voronoi_j}^{old}} \right), \quad \forall j \in [1..C_{connected}];$
 $\tau_{(C_{connected}+1)}^{(C_{connected}+Epoch_{connection})} = - \sum_{j=1}^{C_{connected}} \Delta \tau_j^{(C_{connected}+1)}.$

- una complessità di apprendimento, espressa in numero di operazioni floating-point, pari a:

$$O(Epoch_{connection} \cdot C^2 \cdot M)$$

- il comportamento, in fase di apprendimento, schematizzato in figura 21.

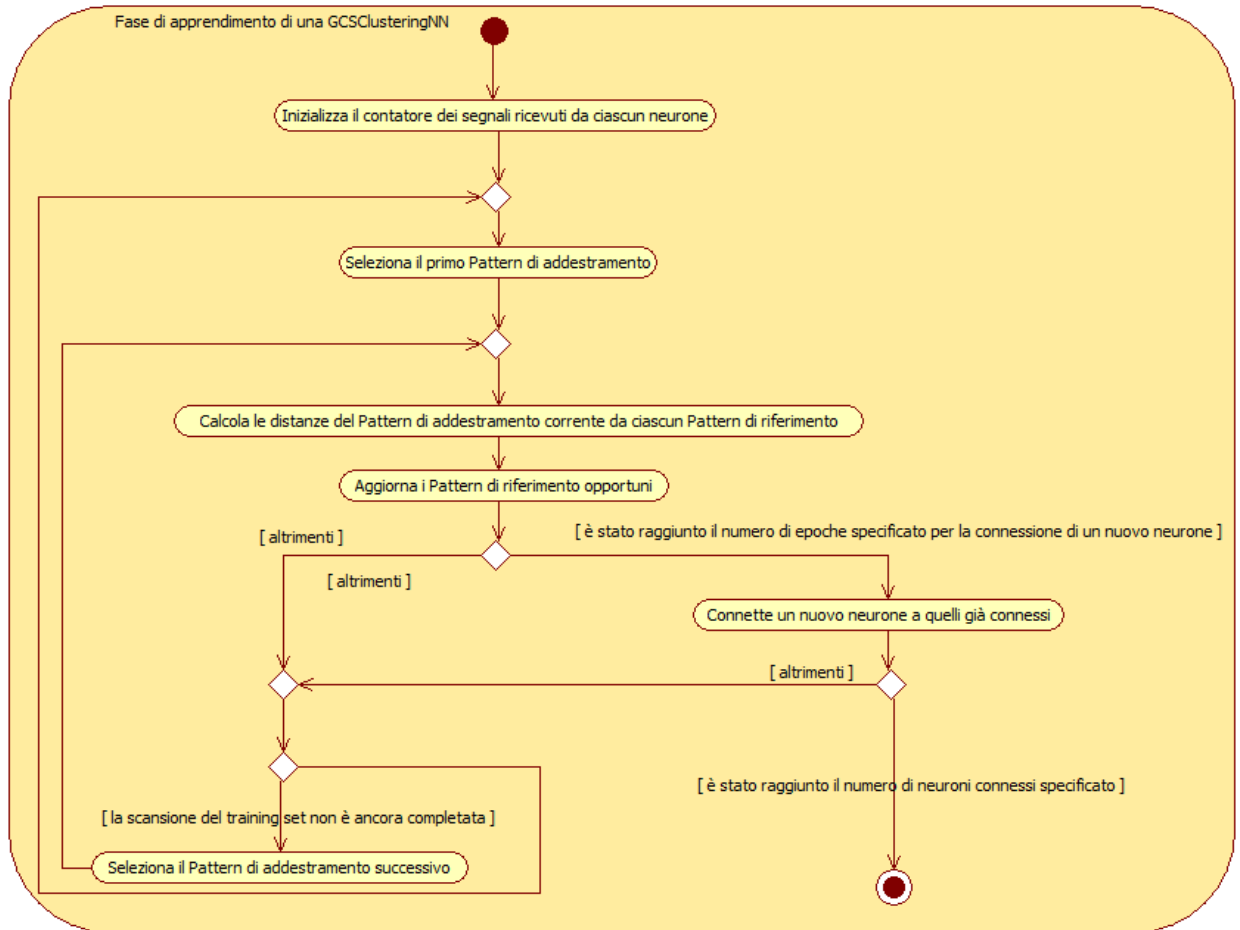


figura 21: Diagramma delle Attività della fase di apprendimento per una GCSClustering

2.3.4 Reti neurali non supervisionate multi-livello per il clustering

Se si esaminano, attentamente le formule per l'aggiornamento dei Pattern di riferimento, per le MaximumEntropyClusteringNN, NeuralGasClusteringNN, SOMClusteringNN e GCSClusteringNN, si può osservare che esse possono riguardare più di un neurone. Tali reti sono

dette avere un apprendimento *softmax*. Il rischio principale derivante dall'uso di reti con apprendimento *softmax* è il *sotto-dimensionamento della rete*, con conseguente *insufficiente capacità di generalizzazione*.

Definizione di sotto-dimensionamento di una rete per la classificazione di Pattern. Si ha il *sottodimensionamento di una rete per la classificazione di Pattern* quando il numero di neuroni della rete è pari a quello delle classi desiderate e quest'ultimo è troppo basso. In tal caso al termine dell'addestramento, si possono ottenere Pattern di riferimento troppo vicini tra loro, che danno luogo a classi sovrapposte.

Una soluzione a tale problema, è rappresentata dalle reti neurali multi-livello. Tali reti risolvono, inoltre, il problema del *labeling* (etichettatura) manuale dei neuroni al termine dell'apprendimento di una ClusteringNN per associare ad ogni neurone un'etichetta di classe, tenendo conto del fatto che non sempre tutti i neuroni della rete saranno ricettivi (si veda il Capitolo successivo).

Una MultiLayerClusteringNN è una rete neurale multi-livello per il ClusteringNN ha le seguenti proprietà:

- è una struttura *composta* da uno o più livelli (si veda figura 22);
- offre le stesse funzionalità di base di una generica ClusteringNN: addestramento e clustering;
- ciascun livello *può essere configurato per usare un tipo differente di ClusteringNN* (come schematizzato in figura 25): tutti i livelli di indice superiore, devono avere un numero di neuroni inferiore;
- *la lunghezza del Pattern di input e il training set sono uguali per tutti i livelli*;
- il comportamento in fase di addestramento è schematizzato in figura 23;
- il comportamento per individuare il cluster di appartenenza del Pattern di input è schematizzato in figura 24;
- data una distribuzione e un'impostazione di parametri di configurazione, o sbaglia sempre o non sbaglia mai la rappresentazione dei dati (si veda il Capitolo successivo).

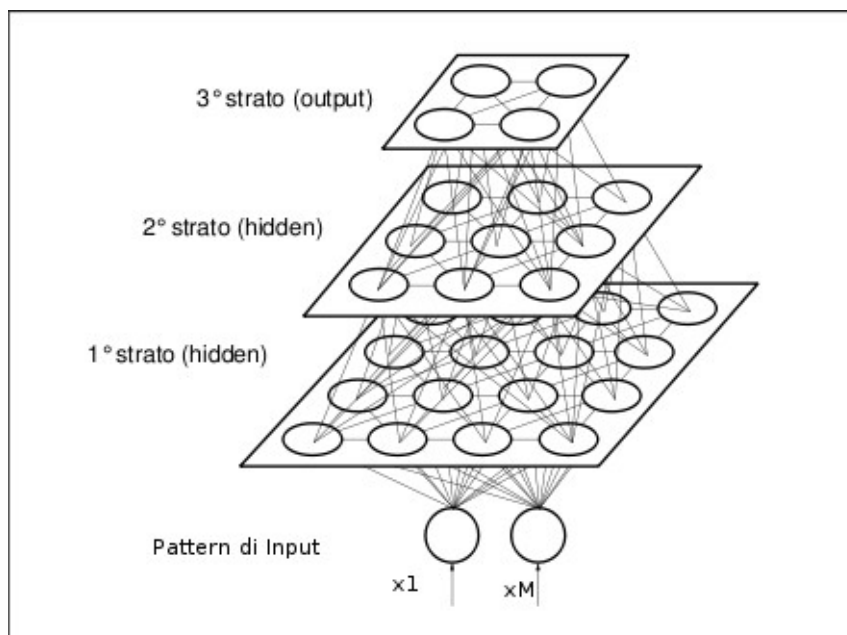


figura 22: Struttura di una MultiLayerClusteringNN

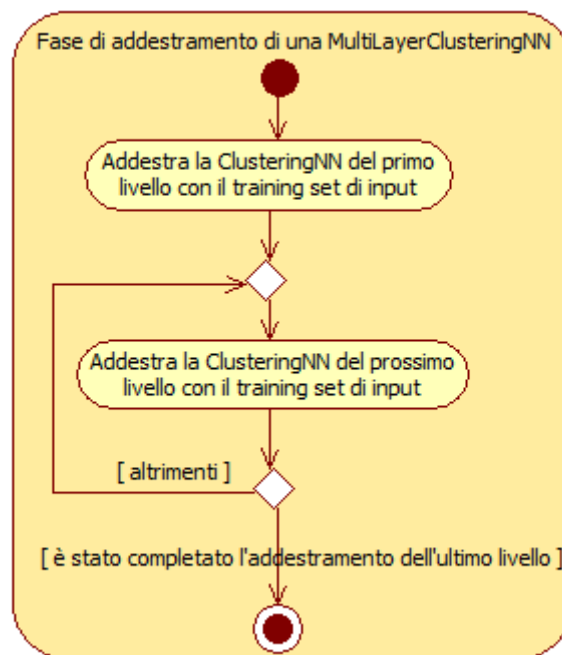


figura 23: Diagramma delle Attività per la fase di addestramento di una MultiLayerClusteringNN

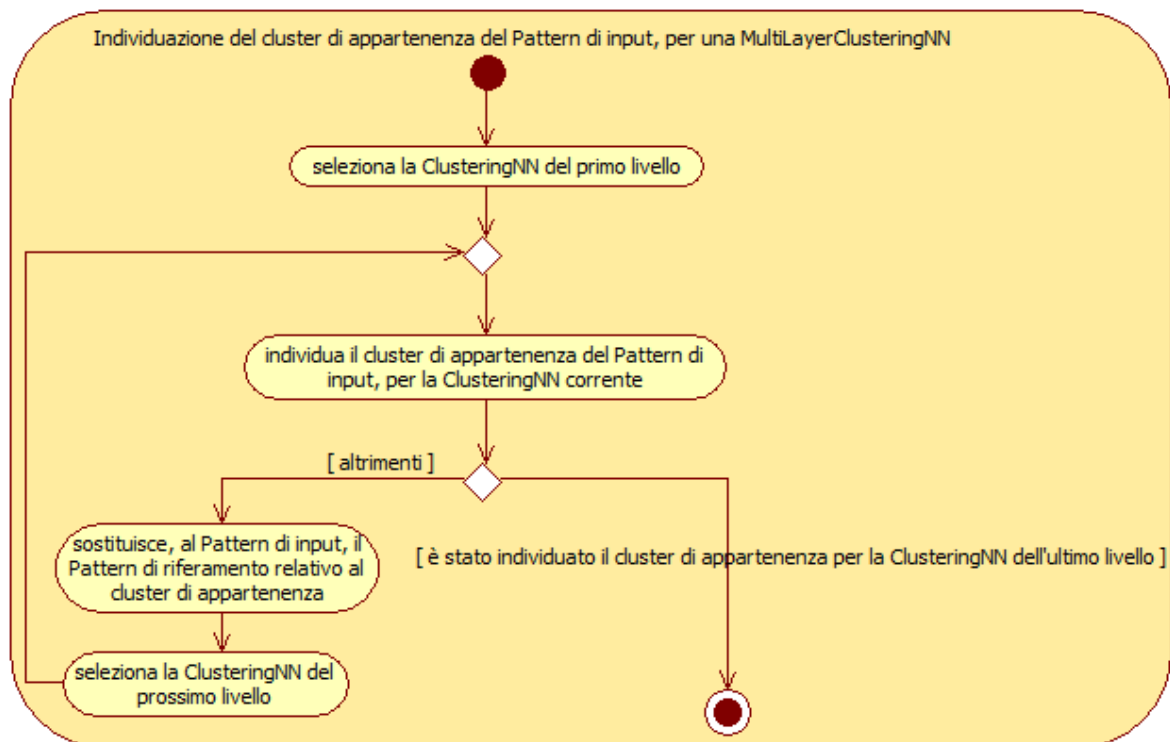


figura 24: Diagramma delle Attività per l'individuazione del cluster di appartenenza di un Pattern di input, per una MultiLayerClusteringNN

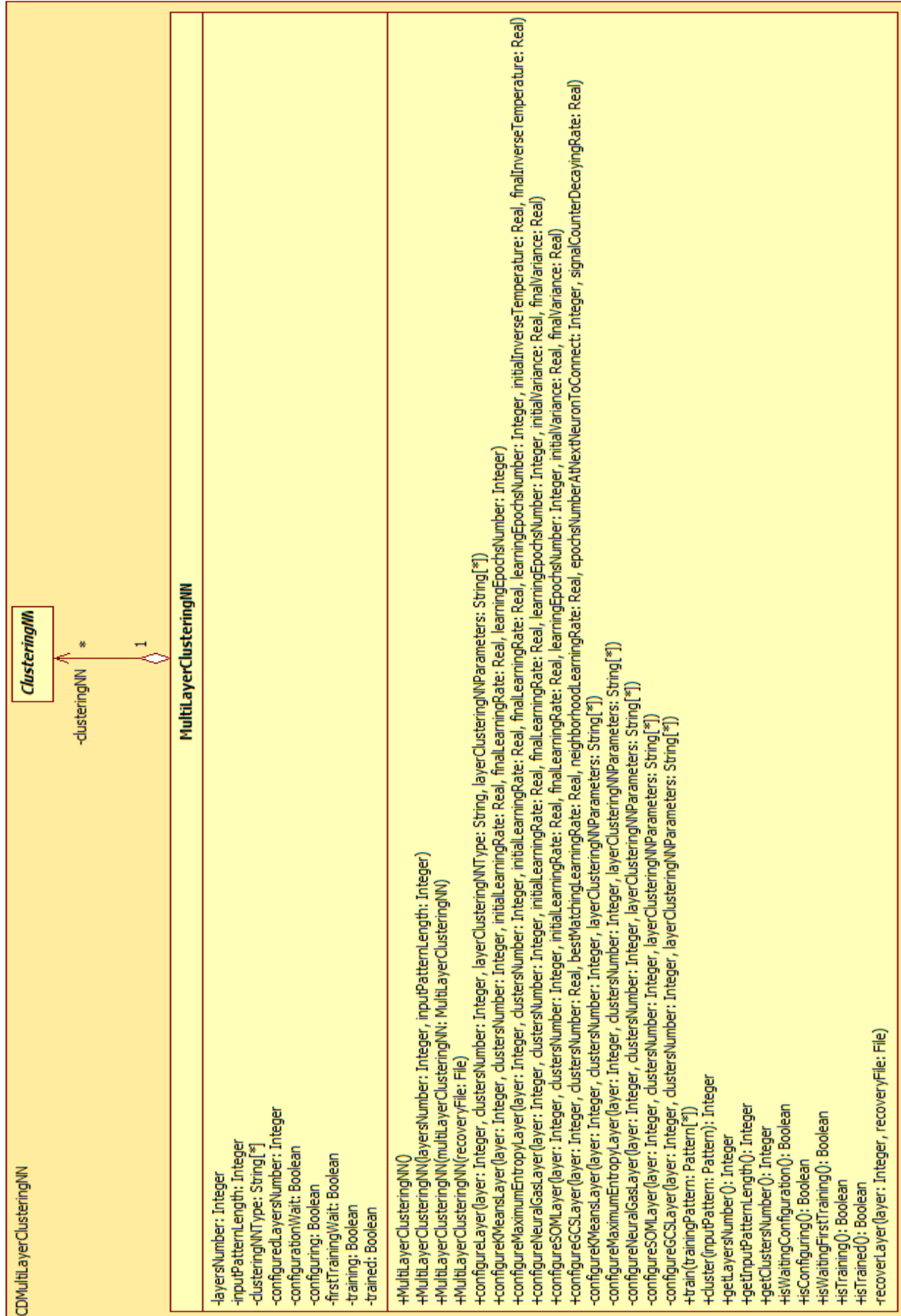


figura 25: Diagramma delle classi che illustra la relazione tra MultiLayerClusteringNN e ClusteringNN.

2.3.5 Estrazione delle finestre significative

In generale la segmentazione di un'immagine potrebbe essere effettuata con il clustering dell'insieme di tutti i suoi Pixel. Tuttavia questa procedura sarebbe computazionalmente onerosa e non terrebbe conto del fatto che, in un'immagine a largo campo, l'informazione statistica sui Pixel è altamente ridondante.

Per queste ragioni si è scelto di effettuare un campionamento statisticamente robusto dell'immagine, attraverso l'estrazione di “finestre” che rappresentino, in modo completo, l'intervallo dinamico delle luminosità dei Pixel, e che comprendano regioni rappresentative dell'immagine tanto del fondo quanto degli oggetti.

Una finestra (Window) significativa è

- una zona (di area quadrata) contenente oggetti, la cui luminosità cada all'interno di un opportuno intervallo;
- con lati di uguale lunghezza;
- individuata da due coppie di coordinate (WindowCoordinates) (x_1, y_1) e (x_2, y_2) , rispettivamente del primo Pixel nell'angolo superiore sinistro e dell'ultimo Pixel nell'angolo inferiore destro;

NExt2IS può ottenere del tale insieme di finestre significative in uno dei due seguenti modi:

- il Client lo specifica direttamente (o mediante un file in formato ds9), richiedendo di fatto la segmentazione semi-automatica dell'immagine di input. In tal caso per ottenere una buona segmentazione è fondamentale l'esperienza dell'Utente (operatore umano) che usa il Client;
- NExt2IS lo estrae automaticamente, usando un estrattore di finestre significative (MeaningFullWindowsExtractor): questo nel caso in cui il Client richieda la segmentazione automatica dell'immagine di input.

Un estrattore di finestre significative, MeaningFullWindowsExtractor, ha le seguenti proprietà:

1. il *numero di finestre* da estrarre da una selezione di finestre significative dell'immagine di input;
2. la *lunghezza del lato* di una finestra significativa da estrarre.

Un tipo di MeaningFullWindowsExtractor (come schematizzato in figura 26) sono gli HistogramBasedMeaningFullWindowsExtractor, che basano l'estrazione delle finestre significative sullo studio dell'istogramma dell'intervallo dinamico (cioè delle luminosità dei Pixel) dell'immagine.

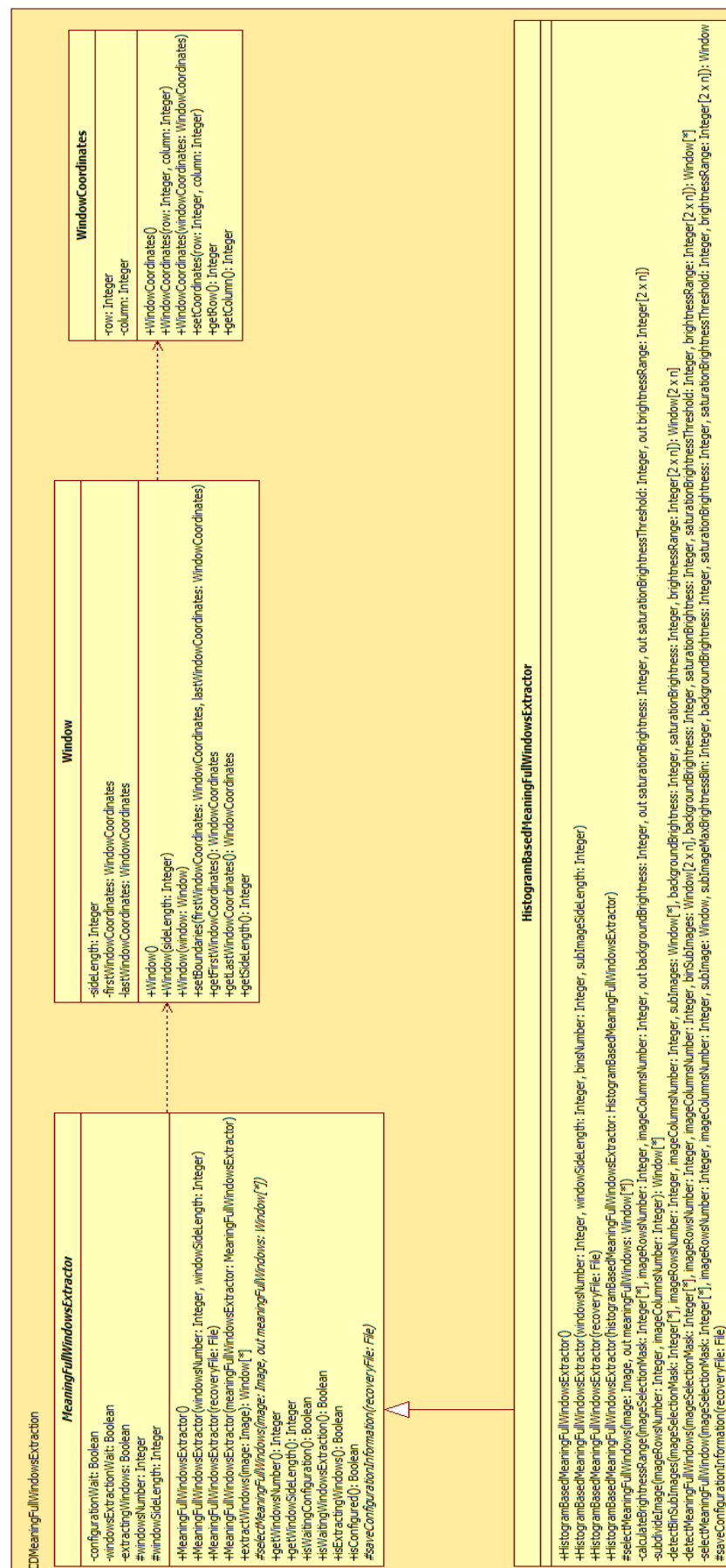


figura 26: Diagramma delle classi per la gerarchia e il funzionamento di un MeaningFullWindowsExtractor

Estrattore di finestre significative basato sul metodo dell'istogramma

Un `HistogramBasedMeaningFullWindowsExtractor` è un `MeaningFullWindowsExtractor` con le seguenti proprietà:

- un *numero di partizioni (bin)* in cui suddividere l'intervallo di luminosità in cui cade la maggior parte dei Pixel dell'immagine, escludendo quelli con luminosità troppo elevata, o debole;
- la *lunghezza del lato* di una sotto-immagine, dell'immagine di input, per facilitare il rilevamento e la selezione dell'insieme di finestre significative, da cui verranno estratte finestre per un numero minore o al più uguale a quello specificato;
- un comportamento, in fase di selezione delle finestre significative, come quello schematizzato in figura 27.

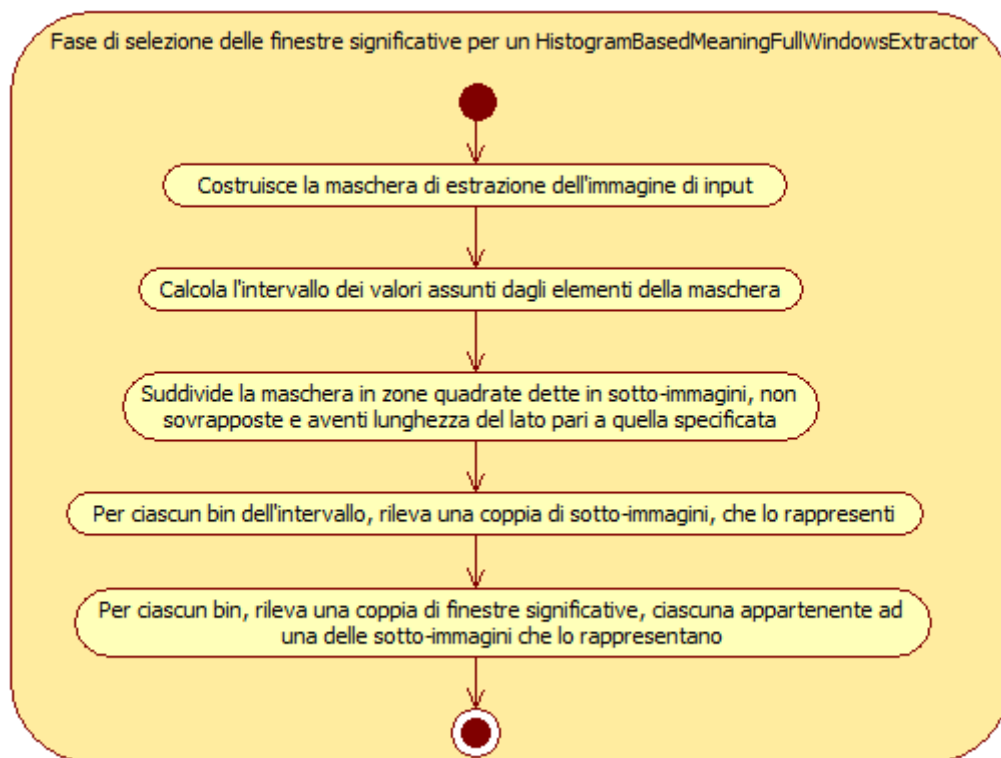


figura 27: Diagramma delle Attività della fase di selezione delle finestre significative per un `HistogramBasedMeaningFullWindowsExtractor`

dove:

- la *maschera di selezione* $Mask_{sel}$, ha le stesse dimensioni dell'immagine di input (Image: attualmente sono ammesse solo immagini mono-banda) ed è tale che, detti IR e IC, rispettivamente il numero di righe e colonne dell'immagine:

$$Mask_{sel}(i, j) = parte\ intera(Image(i, j)) \forall i \in [1..IR] \ e \ j \in [1..IC]$$

- l'intervallo dei valori assunti dagli elementi validi di $Mask_{sel}$ (il cui calcolo è

2 La segmentazione di un'immagine astronomica

schematizzato in dettaglio in figura 28) ha le seguenti proprietà:

- un valore minimo, $lum_{background}$;
- un valore massimo, $lum_{saturation}$;
- un valore soglia, $lum_{saturationThreshold}$, in base al quale, nel rilevamento delle finestre significative, sono scartate quelle con almeno un elemento con valore superiore ad esso;
- è partizionato in sottointervalli disgiunti (bin), tali che in ciascuno di essi (tranne, eventualmente l'ultimo) con all'incirca lo stesso numero di elementi di $Mask_{sel}$;
- le due sotto-immagini rappresentative di un bin sono quelle con il maggior numero di elementi di $Mask_{sel}$ ad esso appartenenti;
- la finestra significativa rilevata, per ciascun sotto-immagine rappresentativa di un bin, ha le seguenti proprietà:
 - si ottiene scandendo la sotto-immagine dall'alto verso il basso, e da sinistra verso destra;
 - contiene il maggior numero di elementi di $Mask_{sel}$ appartenenti dal bin correntemente rappresentato dalla sotto-immagine di appartenenza;
- la complessità per la selezione delle finestre significative, espressa in numero di operazioni floating-point, è:

$$O((IR \times IC) \log(IR \times IC) + (2 \cdot tot_{bin} \times l_{subImage}^2) \times (SIR \times SIC + l_{window}^2))$$

dove:

- tot_{bin} è il numero di bin usato per il partizionamento dell'intervallo $[lum_{background} .. lum_{saturation}]$;
- $l_{subImage}$ è la lunghezza del lato per una sotto-immagine;
- l_{window} è la lunghezza del lato per una finestra;
- SIR e SIC sono, rispettivamente, il numero di righe e di colonne per la suddivisione di $Mask_{sel}$ in sotto-immagini;
- IR e IC sono, rispettivamente, il numero di righe e di colonne dell'immagine di input (e, quindi, di $Mask_{sel}$);
- il numero delle finestre significative effettivamente estratte può essere minore di quello specificato, nel caso in cui il numero delle finestre significative selezionate per l'estrazione sia inferiore ad esso.

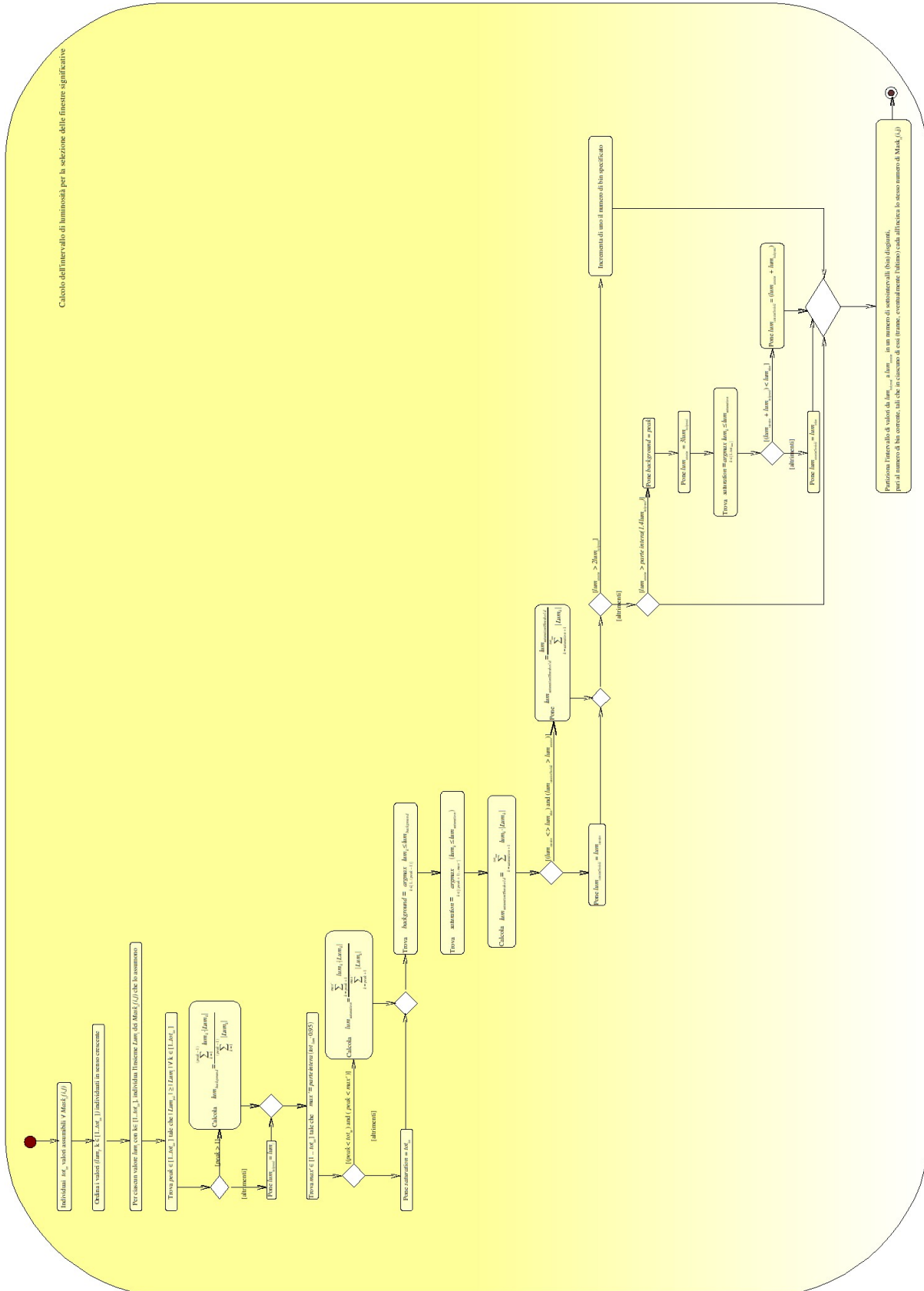


figura 28: Schema per il calcolo dell'intervallo di valori, per un *HistogramBasedMeaningFullWindowsExtractor*

2.4 La Costruzione della Maschera di Segmentazione

La costruzione della *maschera di segmentazione* si articola nei seguenti passi:

1. si crea una matrice di interi delle stesse dimensioni dell'immagine di input;
 2. a ciascuno degli elementi di tale *maschera* viene assegnato il valore intero, decrementato di uno, che identifica il cluster di appartenenza del Pixel corrispondente,
- ed ha una complessità, in numero di operazioni floating-point pari a:

$$O(IR \times IC)$$

2.5 L'Ordinamento dei Pixel

L'etichettatura dei Pixel dell'immagine di input, dato il loro partizionamento (clustering), si articola nei seguenti passi:

1. per ogni cluster $S_j \in [1..C]$ si calcola la luminosità media, mb_j , dei Pixel, ad esso appartenenti;
 2. si stila una classifica delle luminosità medie calcolate, dalla più piccola alla più grande di esse. Le posizioni della classifica vanno da 0 a $(C - 1)$. La classe 0 è composta dai Pixel che riconosciuti come appartenenti allo sfondo dell'immagine di input;
 3. ad ogni cluster $S_j \in [1..C]$ viene associata una classe, la cui etichetta è la pozione occupata da mb_j nella classifica stilata;
 4. gli $(n - 1) / 2$ Pixel per bordo, dell'immagine di input, che sono stati esclusi a priori dalla fase di Clustering, appartengono alla classe 0,
- ed ha una complessità, in numero di operazioni floating-point, pari a:

$$O((IR \times IC) + C \log(C))$$

In tal modo la prima maschera prodotta contiene pixel che assumono un numero di valori pari al numero di cluster, con luminosità crescente, e quindi facilmente interpretabile dall'utente.

Succesivamente può essere prodotta una maschera binaria di segmentazione, attribuendo al fondo i pixel di classe 0 e agli oggetti tutti gli altri.

La procedura di determinazione della maschera di segmentazione richiede infine l'assegnazione, a ciascuno dei suoi elementi dell'etichetta della classe di appartenenza del Pixel corrispondente. Questa procedura ha una complessità, in numero di operazioni floating-point, pari a:

$$O(IR \times IC)$$

2.6 La Configurazione

NExt2IS può essere configurato, principalmente, in uno dei due seguenti modi:

1. per la segmentazione semi-automatica, in tal caso si specificano, eventualmente mediante un file:

(I) la configurazione per la FeaturesExtractionNN da usare, cioè:

(A) i parametri base: lunghezza del Pattern di input, numero delle caratteristiche da

estrarre;

(B)il nome del tipo di FeaturesExtractionNN da usare: StandardPCANN, RobustPCANN o NonLinearPCANN;

(C)la lista di parametri specifica per il tipo di FeaturesExtractionNN scelto;

(II)la configurazione per la MultiLayerClusteringNN da usare, cioè:

(A)il livello da configurare: bottom, middle, top;

(B)il numero di cluster da usare per il livello specificato;

(C)il nome del tipo di ClusteringNN da usare: KmeansClusteringNN, MaximumEntropyClusteringNN, NeuralGasClusteringNN, SOMClusteringNN o GCSClusteringNN;

(D)la lista di parametri specifici per il tipo di ClusteringNN scelto;

2.per la segmentazione automatica, in tal caso si specificano, eventualmente mediante un file, oltre alle configurazioni per FeaturesExtractionNN e MultiLayerClusteringNN, anche quella per il MeaningFullWindowsExtractor, e cioè:

(I)i parametri base: numero delle finestre significative da estrarre, e lunghezza del lato di una finestra;

(II) il nome del tipo di MeaningFullWindowsExtractor da usare (attualmente solo HistogramBasedMeaningFullWindowsExtractor);

(III)la lista dei parametri specifici per il MeaningFullWindowsExtractor scelto.

Le informazioni, non contenute in un file di configurazione, riguardanti i parametri specifici di un determinato tipo di componente, hanno la forma “<nomeParametroSpecifico> *spazio* <valoreParametroSpecifico>”.

In appendice sono riportati alcuni esempi di file di configurazione.

2.7 Il Salvataggio dei Risultati Intermedi

Come accennato in precedenza, il Client può richiedere a NExt2IS di salvare i risultati intermedi di una segmentazione in uno o più file. I file usati per il salvataggio dei risultati intermedi sono i seguenti:

1.un file .reg che contiene le coordinate finestre significative, estratte dall'immagine di input. Tale file (leggibile dal programma ds9 per la visualizzazione delle finestre) viene fornito in output solo su richiesta esplicita del Client, a patto che quest'ultimo abbia selezionato la segmentazione automatica;

2.tre file contenenti le informazioni per un eventuale ripristino della segmentazione:

(I)recoverySegmentationFromFeaturesExtractionNNTraining, contenente le informazioni per ripristinare la segmentazione automatica dall'addestramento delle FeaturesExtractionNN fino alla fine;

(II)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine;

(III) recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine,

fino alla fine;

2. un file, `segmentedImage.fits` contenete il risultato dell'applicazione sull'immagine di input della propria maschera di segmentazione.

In appendice sono riportati alcuni esempi di file per il salvataggio di risultati intermedi.

2.7.1 Esempio di segmentazione

L'immagine, mostrata in figura 30, è il risultato della segmentazione dell'immagine di input, mostrata in figura 29, con il file di configurazione per la segmentazione automatica in appendice.



figura 29: Immagine di input

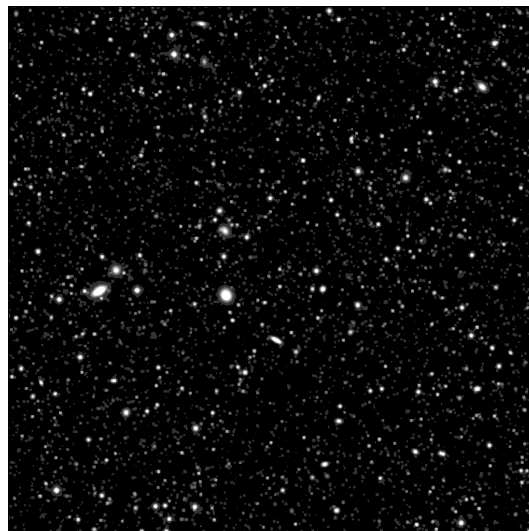


figura 30: Immagine di input segmentata

2.8 Il Sistema NExt2IS e i suoi Sotto-Sistemi

Quanto descritto nei precedenti paragrafi del presente capitolo consente di comprendere la schematizzazione fondamentale del segmentatore NExt2IS, che riportiamo per questioni di leggibilità al termine del capitolo. In particolare si possono individuare due schematizzazioni:

1. la prima (figura 31) illustra in modo sintetico le relazioni che intercorrono tra la classe NExt2IS e i suoi componenti costitutivi;
2. la seconda (figura 32) illustra, sempre in modo sintetico, la scomposizione del Sistema NExt2IS nei sotto-sistemi sviluppati.

2 La segmentazione di un'immagine astronomica



figura 31: Diagramma delle classi di Next2IS

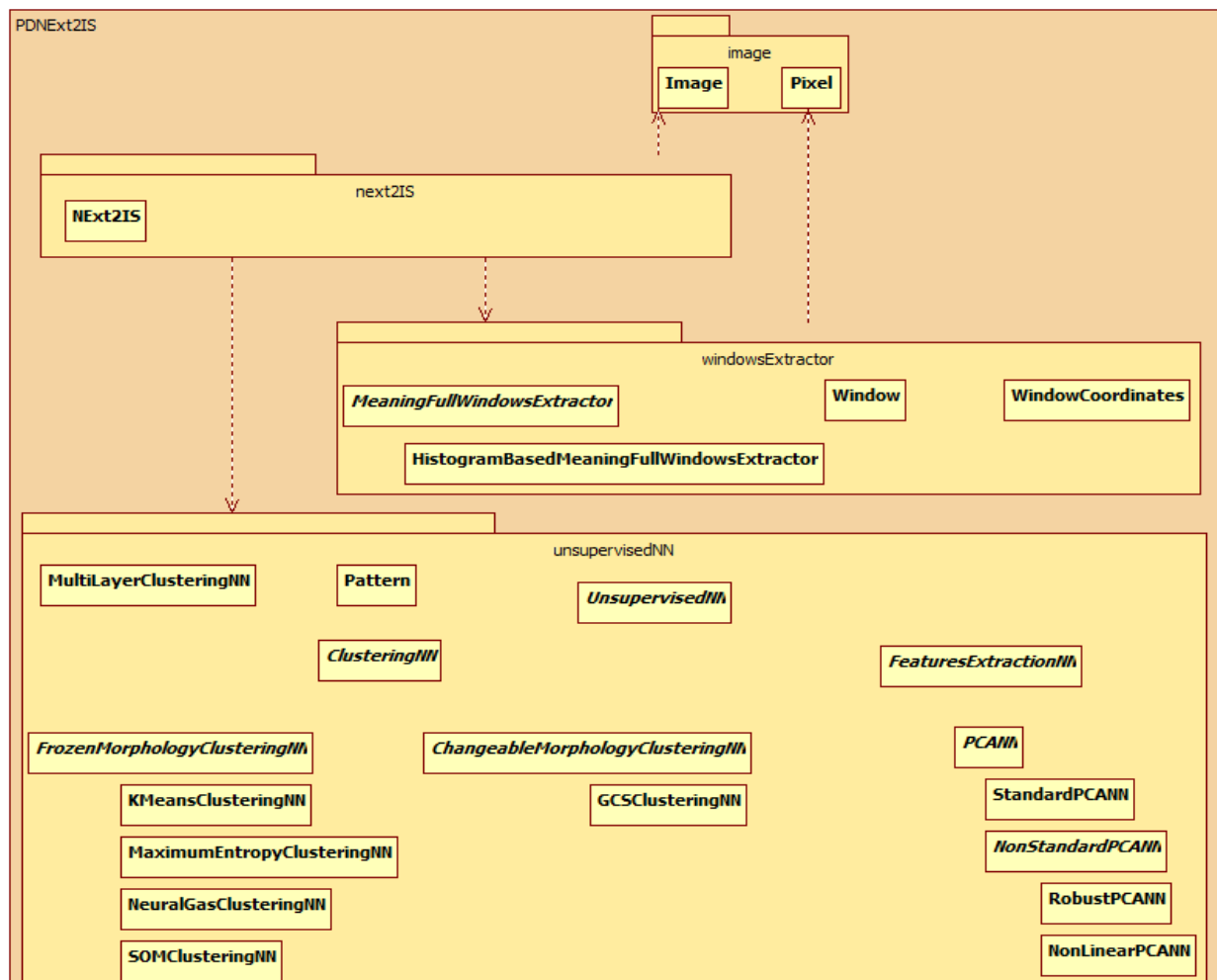


figura 32: Diagramma dei Package rappresentante la scomposizione in Sotto-Sistemi del Sistema Next2IS

2.9 Riferimenti Bibliografici

Longo G., Tagliaferri R., Andreon S.

NExt (Neural Extractor): a new automated tool to extract catalogues from astronomical images
in MINING THE SKY, Proceedings of MPA/ESO/MPE Joint Astronomy Conference held in
Garching,
July 31 – August 4, 2001, (pp. 379-385).

Andreon S., Gargiulo G., Longo G., Tagliaferri R. e Capuano N.

*Wide Field imaging – I. Applications of neural networks to object detection and star/galaxy
classification*
2000 RAS, MNRAS 319, 700-716.

Capuano N.

Classificazione Stelle/Galassie tramite reti neurali.

Tesi di Laurea in Scienze dell'Informazione, presso la Facoltà di Scienze MM. FF. NN.
dell'Università degli Studi di Salerno, A. A. 1997/1998, Capitoli 1, 2, 4, 5 e 6.

Rizzo E.

Un tool basato su reti neurali per il riconoscimento di oggetti astronomici

Tesi di Laurea in Informatica, presso la Facoltà di Scienze MM. FF. NN. dell'Università degli Studi
di Salerno, A. A. 2004/2005, Capitolo 1, 2, 3 e 4.

Rojas R.

Neural Nets – A systematic approach

Springer-Verlag, Berlin, 1996.

Bishop C. M.

Neural Networks for Pattern Recognition

Oxford University Press, 2005

2.10 Siti internet di riferimento

www.neuroinformatik.ruhr-uni-bochum.de

3 Esperimenti di segmentazione

Descrizione immagini utilizzate:

Immagine	Dimensioni (numero pixel)	Intervallo dinamico	Descrizione
A	1024 x 1024	18032 - 65535	Simulazione di un campo celeste rumoroso, contenente oggetti più o meno, estesi e brillanti
B	1024 x 1024	1121 - 65535	Simulazione di un campo celeste rumoroso, contenente oggetti, poco estesi e poco brillanti.
C	2048 x 2048	300 - 78661	Campo celeste reale rumoroso, contenente oggetti, più o meno, estesi e brillanti
D	2048 x 2048	192 - 73831	Campo celeste reale rumoroso, contenente oggetti, più o meno, estesi e brillanti
E	1024 x 1024	18691 - 65535	Simulazione di un campo celeste privo di rumore, contenente oggetti più o meno, estesi e brillanti

Tutte le immagini sopra elencate sono mono-banda.



Immagine 1: A



Immagine 3: B



Immagine 2: C



Immagine 4: D



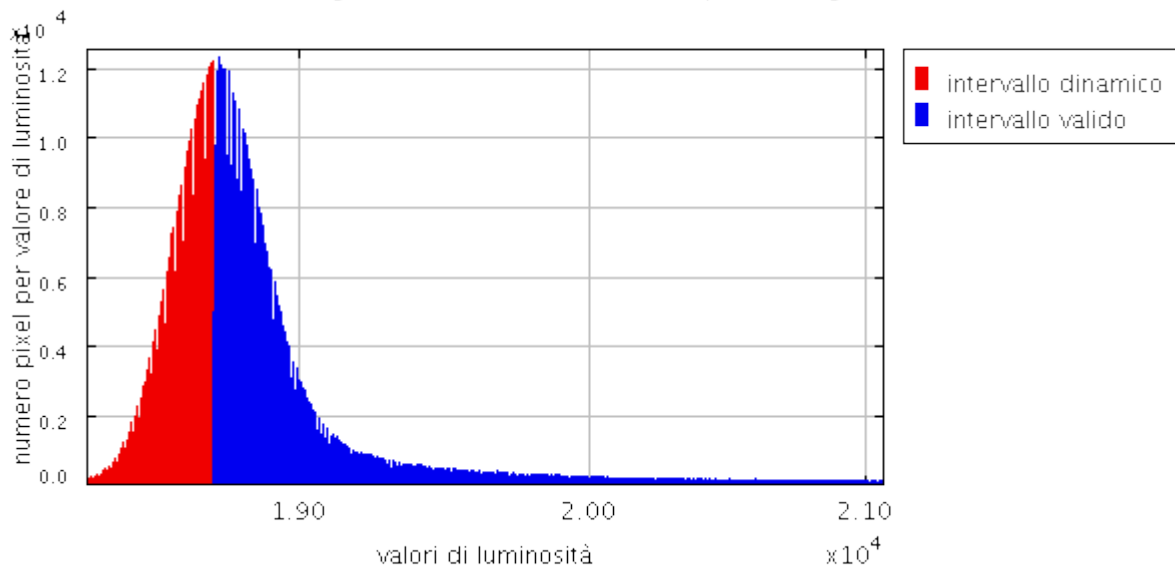
Immagine 5: E

3.1 Estrattore di finestre significative

L'estrazione delle finestre significative dell'immagine, come descritto nel capitolo 2, è eseguita da un'estrattore di finestre significative basato sul metodo dell'istogramma.

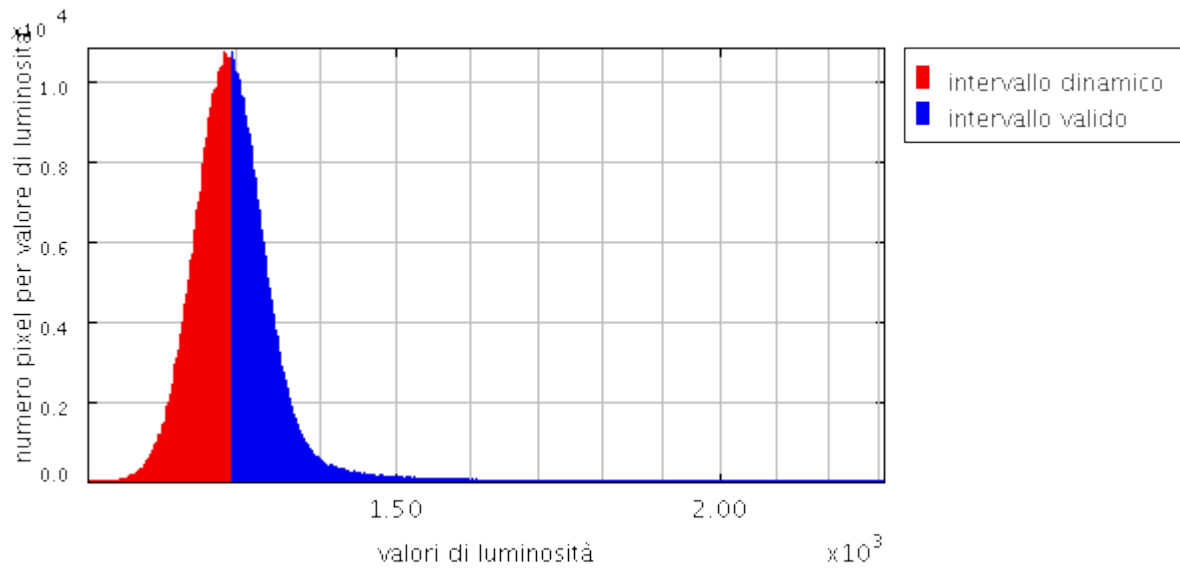
Riportiamo di seguito gli istogrammi relative alle immagini (A, B, C ed E) sopraelencate.

Istogramma dei valori di luminosità per l'immagine A

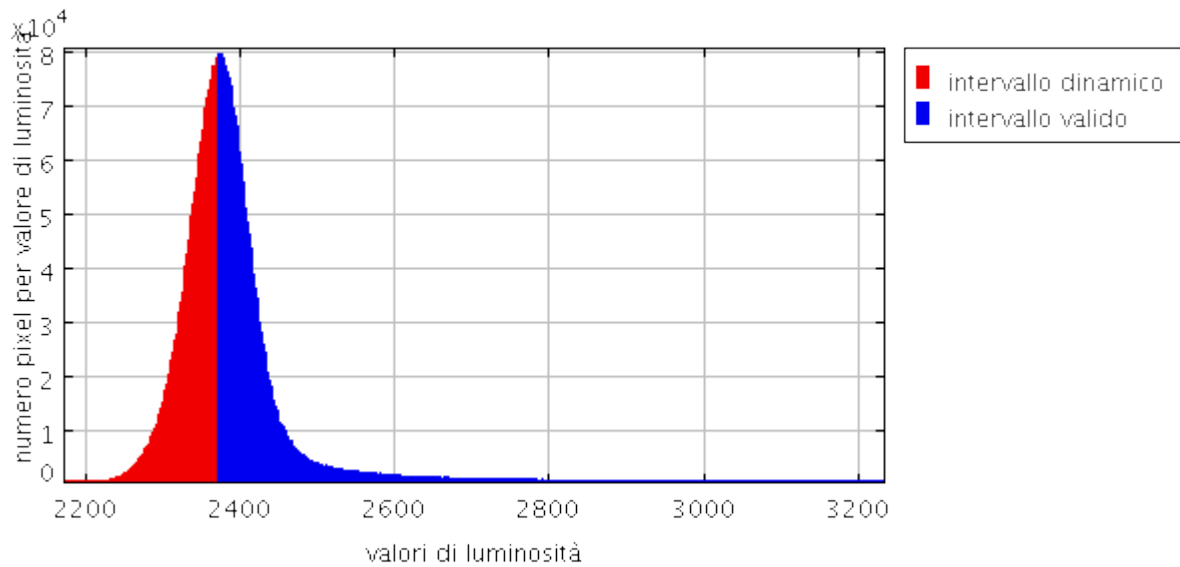


3 Esperimenti di segmentazione

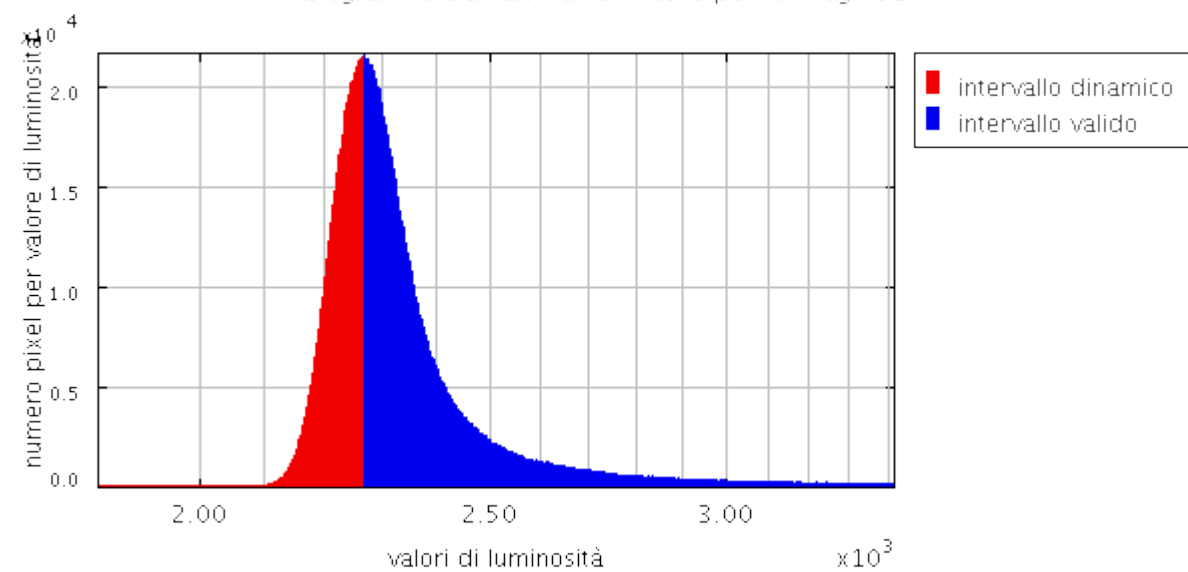
Istogramma dei valori di luminosità per l'immagine B



Istogramma dei valori di luminosità per l'immagine C



Istogramma dei valori di luminosità per l'immagine D



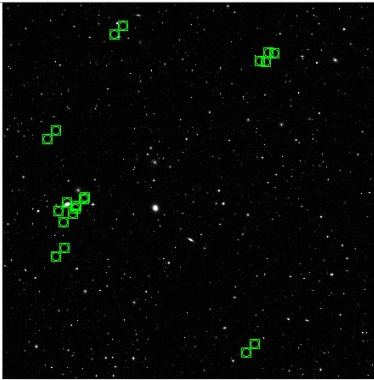
Analizziamo ora il comportamento di NextII e dell'estrattore di finestre significative considerato, per dimensioni diverse delle finestre significative (25 x 25, 50 x 50, 100 x 100). Tutte le segmentazioni sono state ottenute fissando i seguenti parametri di configurazione:

1. Per l'estrattore di finestre significative basate sul metodo dell'istogramma::
 - (I) la lunghezza del lato di una sotto-immagine è 100;
 - (II) il numero delle finestre da estrarre è 10;
 - (III) il numero di partizioni (bin) in cui viene suddiviso l'intervallo dei valori di luminosità è 6.
2. Per l'estrazione delle caratteristiche di un pattern è utilizzata una rete neurale per la PCA:
 - (I) il pattern di input ha lunghezza 9 (quindi viene usato un intorno di diametro 3 pixel);
 - (II) il tipo di PCANN usato è la StandardPCANN;
 - (III) il numero di caratteristiche da estrarre è 3;
 - (IV) il tasso di apprendimento è 0.03;
 - (V) il tipo di feed-back di apprendimento è simmetrico;
 - (VI) la soglia di arresto per l'apprendimento è 0.00001.
3. Per il partizionamento dell'insieme delle caratteristiche estratte è stata utilizzata una MultiLayerClusteringNN a tre livelli:
 - (I) la ClusteringNN del primo livello è una GCSClusteringNN con:
 - A. 50 neuroni;
 - B. un tasso di apprendimento best-matching di 0.1;
 - C. un tasso di apprendimento neighborhood di 0.01;
 - D. un numero di epoche per la connessione di ciascun neurone (dopo il terzo) di 1000;
 - E. un tasso di decadimento dei contatori dei segnali ricevuti dai neuroni, durante l'apprendimento di 0.05;
 - (II) le ClusteringNN del secondo e terzo livello sono due NeuralGasClusteringNN, con rispettivamente 20 e 6 neuroni e:
 - A. un tasso di apprendimento iniziale di 0.7;
 - B. un tasso di apprendimento finale di 0.05;
 - C. una varianza iniziale di, rispettivamente, 3.0 e 2.0;
 - D. una varianza finale di 0.1;
 - E. un numero di epoche di apprendimento pari a 50.

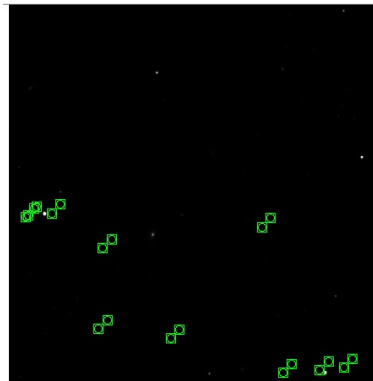
Segmentazione automatica con finestre significative 25 x 25

3 Esperimenti di segmentazione

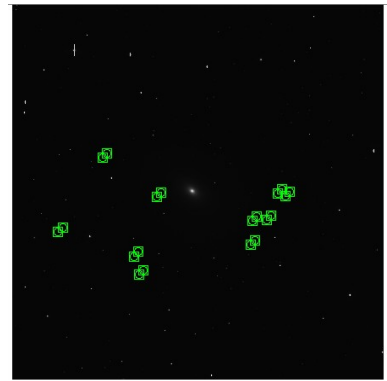
Le seguenti figure mostrano il risultato dell'estrazione automatica delle finestre significative (di dimensione 25x25) delle immagini A, B, C, D ed E. Le coordinate di ciascuna delle finestre significative sono rappresentate dai quadratini verdi.



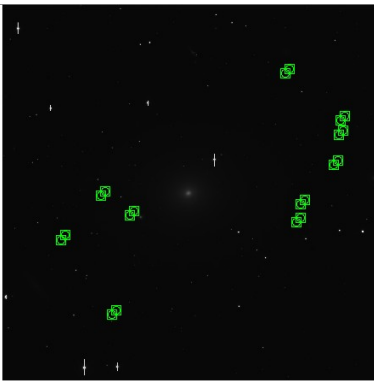
*EstrazioneFinestre 1:
finestre significative 25 x25
dell'immagine A*



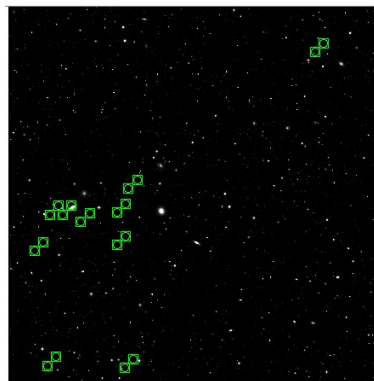
*EstrazioneFinestre 2:
finestre significative 25 x25
dell'immagine B*



*EstrazioneFinestre 3:
finestre significative 25 x25
dell'immagine C*



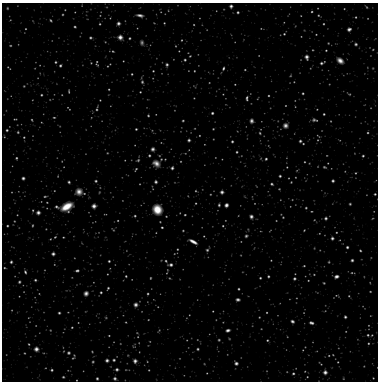
*EstrazioneFinestre 4:
finestre significative 25 x25
dell'immagine D*



*EstrazioneFinestre 5:
finestre significative 25 x25
dell'immagine E*

Le possibili segmentazioni ottenute usando, per ciascuna immagine il proprio insieme di finestre significative estratte, sono le seguenti.

3 Esperimenti di segmentazione



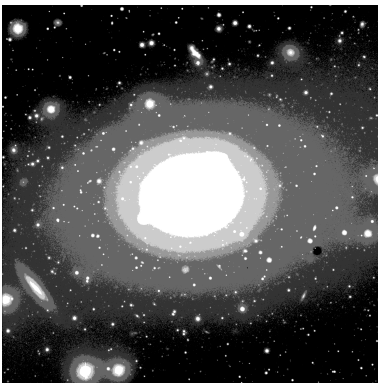
Segmentazione 1: immagine A con finestre significative 25x25



Segmentazione 2: immagine B con finestre significative 25x25



Segmentazione 3: immagine C con finestre significative 25 x 25



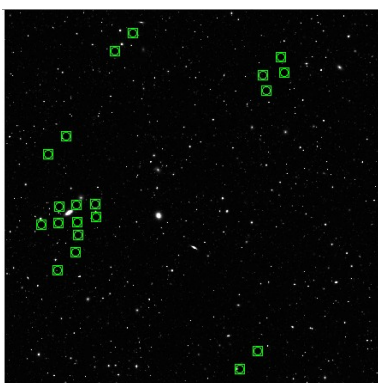
Segmentazione 4: immagine D con finestre significative 25 x 25



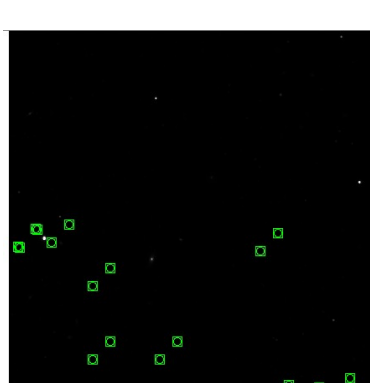
Segmentazione 5: immagine E con finestre significative 25 x 25

Segmentazione automatica con finestre 50 x 50

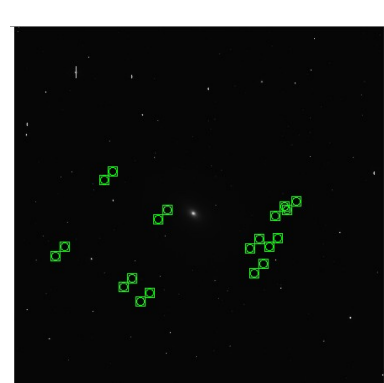
Le seguenti figure mostrano il risultato dell'estrazione automatica delle finestre significative (di dimensione 50x50) delle immagini A, B, C, D ed E. Le coordinate di ciascuna delle finestre significative sono rappresentate dai quadratini verdi.



Estrazione Finestre 6: finestre significative 50 x 50 dell'immagine A

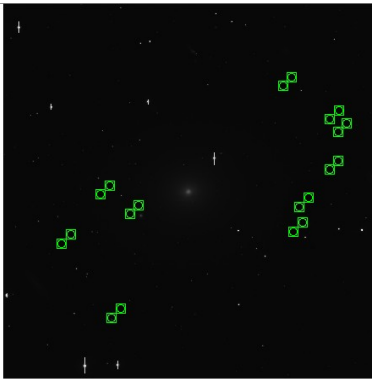


Estrazione Finestre 8: finestre significative 50 x 50 dell'immagine B

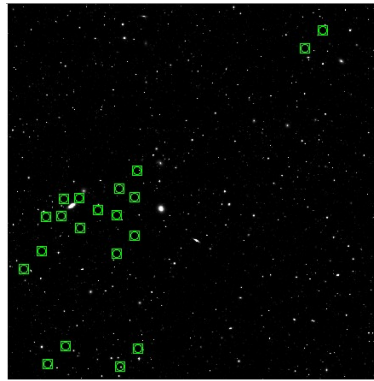


Estrazione Finestre 7: finestre significative 50 x 50 dell'immagine C

3 Esperimenti di segmentazione



*Estrazione Finestre 9:
finestre significative 50 x 50
dell'immagine D*

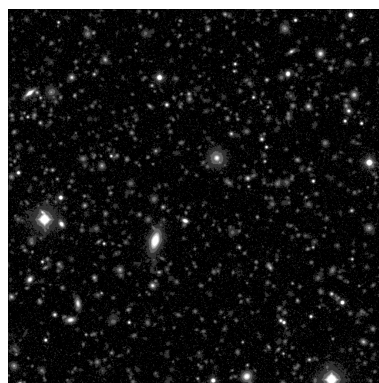


*Estrazione Finestre 10:
finestre significative 50 x 50
dell'immagine E*

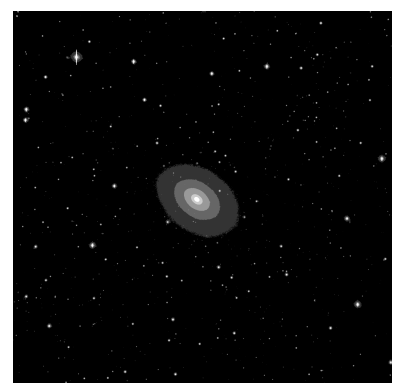
Le possibili segmentazioni ottenute usando, per ciascuna immagine il proprio insieme di finestre significative estratte, sono le seguenti.



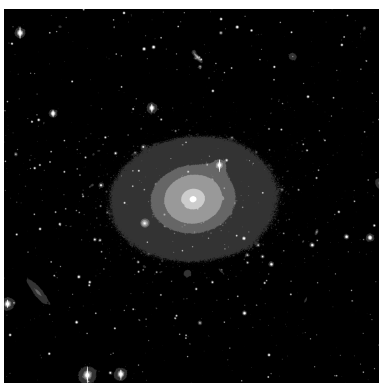
*Segmentazione 6: immagine
A con finestre significative
50x50*



*Segmentazione 7: immagine
B con finestre significative
50x50*



*Segmentazione 8: immagine
C con finestre significative
50x50*



*Segmentazione 9: immagine
D con finestre significative
50x50*



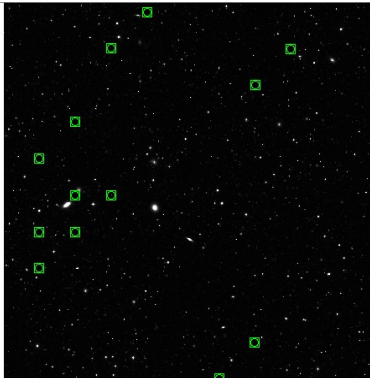
*Segmentazione 10:
immagine E con finestre
significative 50x50*

Segmentazione automatica con finestre 100 x 100

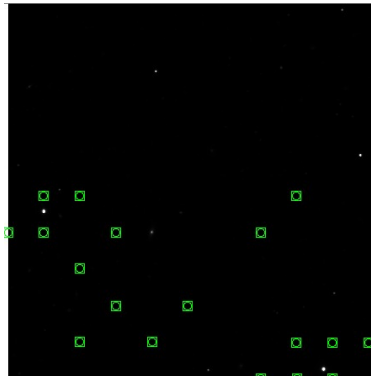
Le seguenti figure mostrano il risultato dell'estrazione automatica delle finestre significative (di

3 Esperimenti di segmentazione

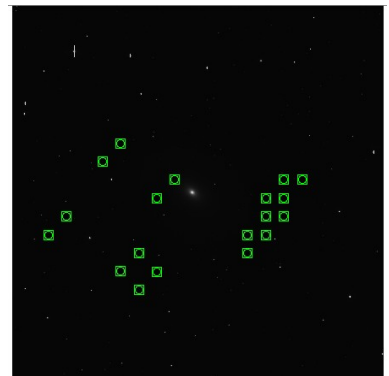
dimensione 100x100) delle immagini A, B, C, D ed E. Le coordinate di ciascuna delle finestre significative sono rappresentate dai quadratini verdi.



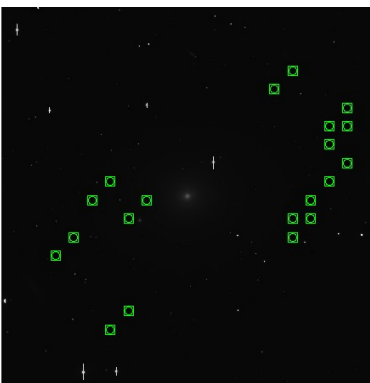
*EstrazioneFinestre 11:
finestre significative 100 x
100 dell'immagine A*



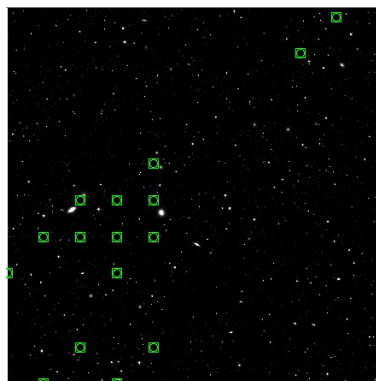
*EstrazioneFinestre 12:
finestre significative 100 x
100 dell'immagine B*



*EstrazioneFinestre 13:
finestre significative 100 x
100 dell'immagine C*

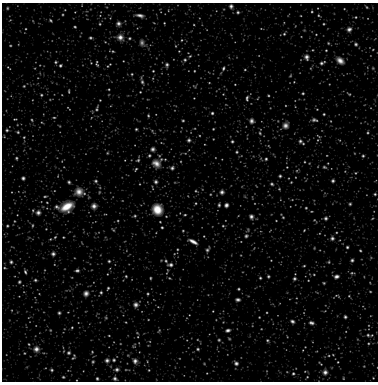


*EstrazioneFinestre 14:
finestre significative 100 x
100 dell'immagine D*

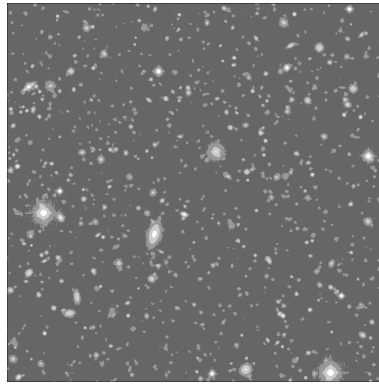


*EstrazioneFinestre 15:
finestre significative 100 x
100 dell'immagine E*

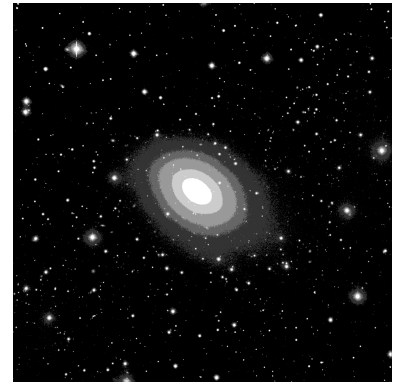
Le possibili segmentazioni ottenute usando, per ciascuna immagine il proprio insieme di finestre significative estratte, sono le seguenti.



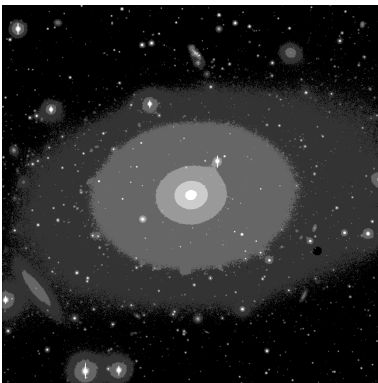
*Segmentazione 11:
immagine A con finestre
significative 100x100*



*Segmentazione 12:
immagine B con finestre
significative 100x100*



*Segmentazione 13:
immagine C con finestre
significative 100x100*



*Segmentazione 14:
immagine D con finestre
significative 100x100*



*Segmentazione 15:
immagine E con finestre
significative 100x100*

Considerazioni

Come si può dedurre, osservando le immagini segmentate:

1. la grandezza dell'immagine può non influire sulla grandezza delle finestre significative;
2. la grandezza delle finestre significative può, in alcuni casi, pregiudicare il risultato della segmentazione.

3.2 Estrazione delle caratteristiche

Come descritto nel capitolo 2, per partizionare l'insieme dei pixel dell'immagine di input, NextII individua il cluster di appartenenza di ciascun pixel tenendo in considerazione il contesto in si trova, e per fare ciò per ogni pixel estrae dall'immagine un pattern contenente quel pixel e i suoi vicini per un intorno di diametro n . All'aumentare del diametro dell'intorno, e del numero di bande del pixel, corrisponde un allungamento del pattern che lo rappresenta, e questo può rallentare, rendendolo poco agevole, il partizionamento dei pixel dell'immagine. Pertanto si ha bisogno di ridurre la lunghezza del pattern, estraendo da esso le caratteristiche rappresentative del suo contenuto informativo. Nel nostro caso le caratteristiche da estrarre da un pattern sono le sue componenti principali, e il metodo usato è basato su l'uso di reti neurali PCA (PCANN).

Un riscontro sull'effettivo funzionamento delle PCANN sviluppate è stato ottenuto, applicando loro un dataset di pattern bi-dimensionali (appartenenti alla bisettrice del primo quadrante del piano

3 Esperimenti di segmentazione

cartesiano) ed estraendo per ciascuno di essi una componente principale, per verificare che il pattern di riferimento $w=(w_1, w_2)$ dell'unico neurone posseduto da ciascuna rete, soddisfi le seguenti condizioni:

- $w_1^2 + w_2^2 = 1;$
- $w_1 + w_2 = \sqrt{2}.$

PCANN		w_1	w_2
Standard	Simmetrica	0.707107	0.707107
	Gerarchica	0.707107	0.707107
Robusta a massimizzazione della varianza	Simmetrica	0.707107	0.707107
	Gerarchica	0.707107	0.707107
Robusta a minimizzazione dell'errore approssimato	Simmetrica	0.707107	0.707107
	Gerarchica	0.707107	0.707107
Robusta a minimizzazione dell'errore non approssimato	Simmetrica	0.707107	0.707107
	Gerarchica	0.707107	0.707107
Non lineare	Simmetrica	0.719075	0.694933
	Gerarchica	0.625565	0.780172

3.3 Pattern Clustering

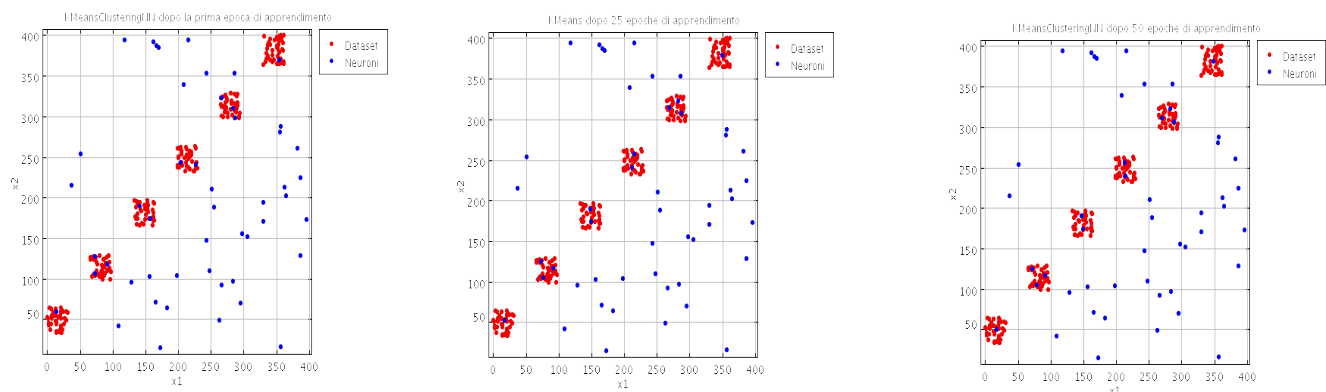
Come descritto nel capitolo 2, per partizionare l'insieme dei pattern rappresentati i pixel dell'immagine di input, NextII usa una rete neurale multi-livello non supervisionata a tre livelli, in cui per ciascun livello può essere usato un differente tipo di ClusteringNN (KMeans, Maximum Entropy, Neural Gas, SOM o GCS). Analizziamo ora il comportamento delle singole ClusteringNN in modo da individuare le più appropriate per ciascun livello: a ciascuna rete sono assegnati 50 e un dataset unico composto da 240 pattern bi-dimensionali, in cui ciascuna componente è un numero intero nell'intervallo da 0 a 400. I 240 pattern appartengono a 6 cluster ben separati.

3.1.1 KMEansClusteringNN

Configurazione rete

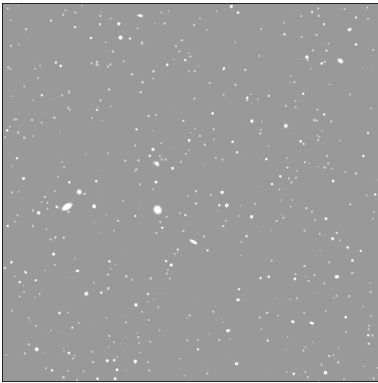
1. lunghezza pattern di input = 2;
2. numero di neuroni = 50;
3. tasso di apprendimento iniziale = 0.7;
4. tasso di apprendimento finale = 0.05;
5. epoche di apprendimento = 50.

Risultato dell'apprendimento

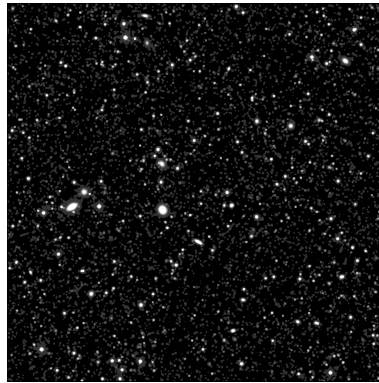


Ricordiamo dal capitolo 2 che, in fase di apprendimento il Kmeans aggiorna solo il pattern di riferimento corrispondente al neurone più vicino a ciascun pattern del training set. Pertanto, come si può vedere dai grafici sovraesposti, molti pattern di riferimento non sono aggiornate. Ne consegue che, al termine dell'apprendimento, per l'effettivo utilizzo della rete saranno ricettivi solo pochi neuroni, rendendola rete inaffidabile, anche se molto veloce. Dunque, questo tipo di rete può essere utilizzata solo nel primo (o, al più secondo) livello, come si può vedere dai seguenti risultati di segmentazione.

3 Esperimenti di segmentazione



Segmentazione 16:
MultiLayerClusteringNN
con 3 livelli KMeans (50, 20,
6 neuroni)



Segmentazione 17:
MultiLayerClusteringNN con
KMeans (a 50 neuroni) e
due livelli di SOM (a 20 e 6
neuroni)



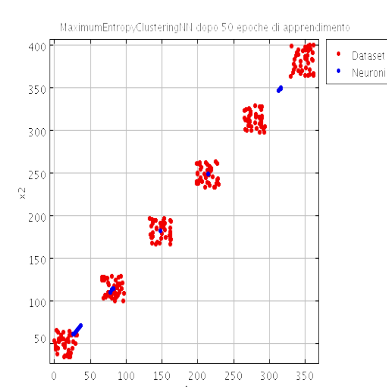
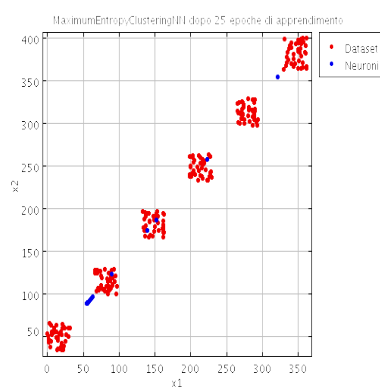
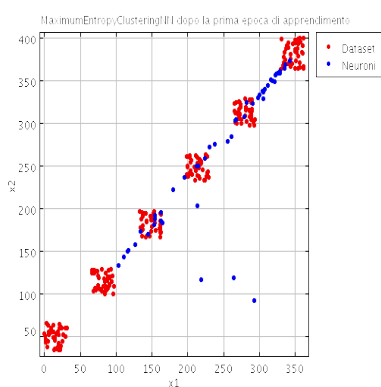
Segmentazione 18:
MultiLayerClusteringNN a
due livelli di KMeans (a 50 e
20 neuroni) e un livello di
NeuralGas (a 6 neuroni)

3.1.2 MaximumEntropyClusteringNN

Configurazione rete

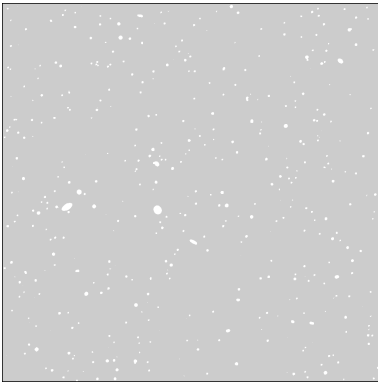
1. lunghezza pattern di input = 2;
2. numero di neuroni = 50;
3. tasso di apprendimento iniziale = 0.7;
4. tasso di apprendimento finale = 0.05;
5. epoche di apprendimento = 50;
6. temperatura inversa iniziale = 5.0;
7. temperatura inversa finale = 100.0.

Risultato dell'apprendimento

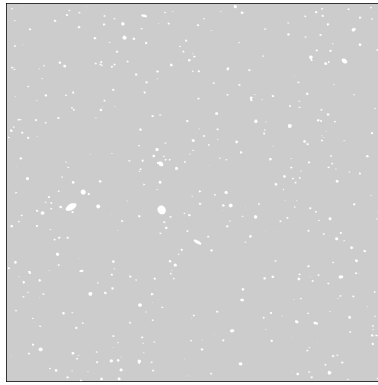


Ricordiamo dal capitolo 2 che, in fase di apprendimento la Maximum Entropy tende a utilizzare tutti i suoi neuroni, purtroppo, come si può vedere dai grafici sovraesposti, i pattern di riferimento tendono a sovrapporsi. Ne consegue che, al termine dell'apprendimento, per l'effettivo utilizzo della rete saranno ricettivi solo pochi neuroni, rendendola rete inaffidabile. Dunque, questo tipo di rete può essere utilizzata al più nel primo livello.

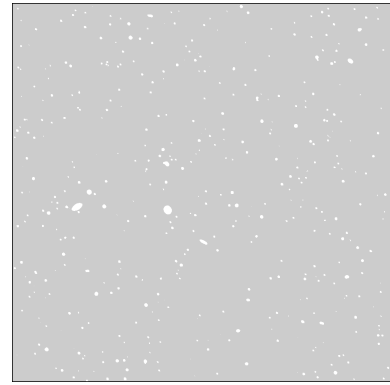
3 Esperimenti di segmentazione



Segmentazione 19:
MultiLayerClusteringNN a 3
livelli di *MaximumEntropy*
(a 50, 20 e 6 neuroni)



Segmentazione 20:
MultiLayerClusteringNN
con *MaximumEntropy* (a 50
neuroni) e due livelli di
NeuralGas (a 20 e 6
neuroni)



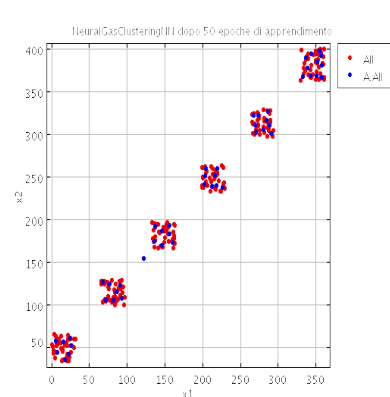
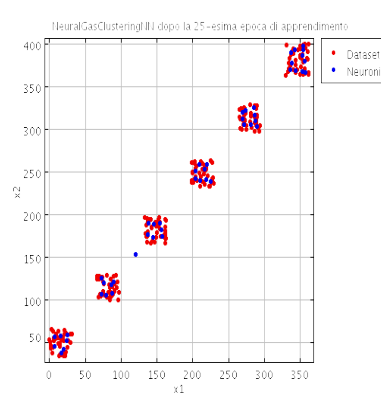
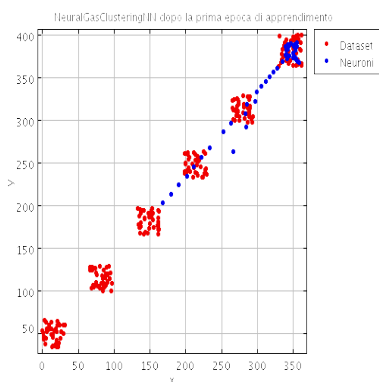
Segmentazione 21:
MultiLayerClusteringNN
con due livelli
MaximumEntropy (a 50 e 20
neuroni) e un livello di *SOM*
(a 6 neuroni)

3.1.3 NeuralGasClusteringNN

Configurazione rete

1. lunghezza pattern di input = 2;
2. numero di neuroni = 50;
3. tasso di apprendimento iniziale = 0.7;
4. tasso di apprendimento finale = 0.05;
5. epoche di apprendimento = 50;
6. varianza iniziale = 10.0;
7. varianza finale = 0.1.

Risultato dell'apprendimento



Ricordiamo dal capitolo 2 che, in fase di apprendimento la Neural Gas tende a utilizzare tutti i suoi neuroni per rappresentare al meglio il training set, raggiungendo una buona rappresentazione dell'input già dopo poche epoche di apprendimento. Dunque, questo tipo di rete può essere

3 Esperimenti di segmentazione

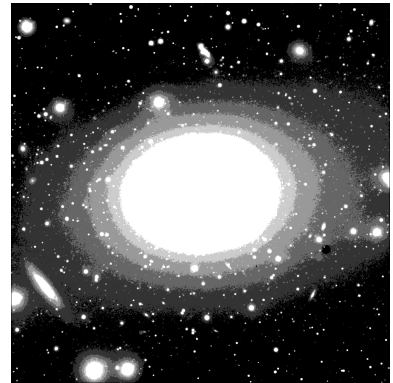
utilizzata per tutti e tre i livelli della multi-livello.



Segmentazione 22:
MultiLayer con tre livelli di
Neural Gas (a 50, 20 e 6
neuroni)



Segmentazione 24:
MultiLayer con tre livelli di
Neural Gas (a 50, 20 e 6
neuroni)



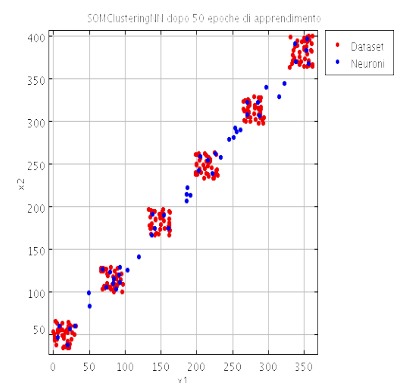
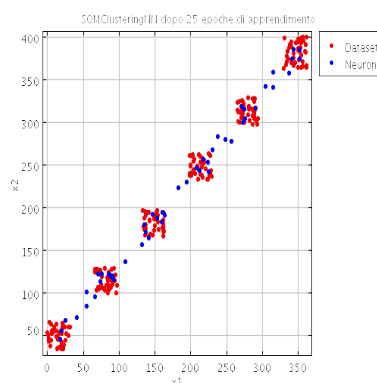
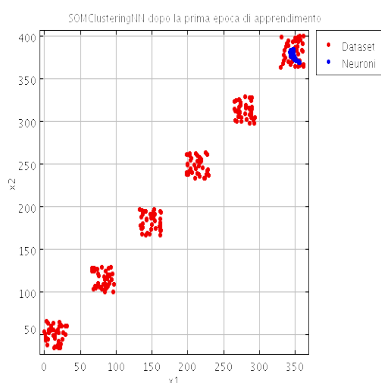
Segmentazione 23:
MultiLayer con tre livelli di
Neural Gas (a 50, 20 e 6
neuroni)

3.1.4 SOMClusteringNN

Configurazione rete

1. lunghezza pattern di input = 2;
2. numero di neuroni = 50;
3. tasso di apprendimento iniziale = 0.7;
4. tasso di apprendimento finale = 0.05;
5. epoche di apprendimento = 50;
6. varianza iniziale = 5.0;
7. varianza finale = 0.1.

Risultati dell'apprendimento



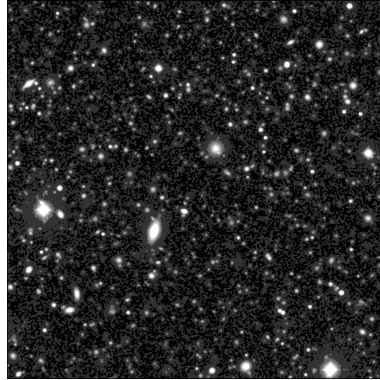
Ricordiamo dal capitolo 2 che, in fase di apprendimento la SOM cerca di assicurare una relazione topologica tra i pattern di riferimento e quelli del training set, muovendo i primi in modo da rispecchiare inizialmente l'organizzazione a griglia dei propri neuroni, per poi farli divergere fino a coprire gli spazi occupati dai pattern di addestramento. Dunque, questo tipo di rete può essere

3 Esperimenti di segmentazione

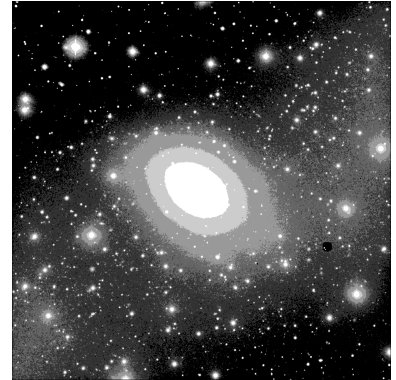
utilizzata per tutti e tre i livelli della multi-livello.



Segmentazione 25:
MultiLayer con tre livelli di
SOM (a 50, 20 e 6 neuroni)



Segmentazione 26:
MultiLayer con tre livelli di
SOM (a 50, 20 e 6 neuroni)



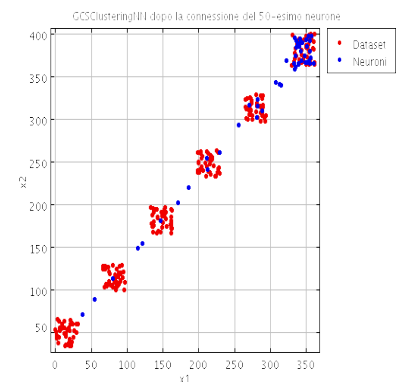
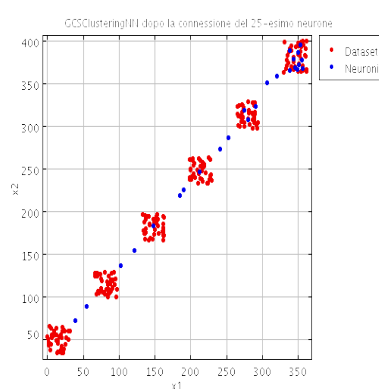
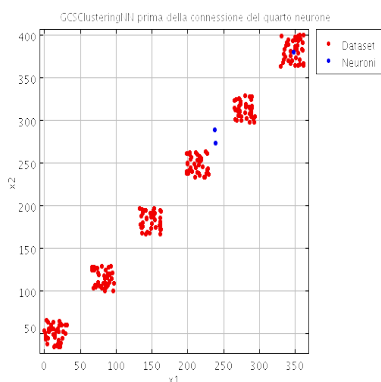
Segmentazione 27:
MultiLayer con tre livelli di
SOM (a 50, 20 e 6 neuroni)

3.1.5 GCSClusteringNN

Configurazione rete

1. lunghezza pattern di input = 2;
2. numero di neuroni = 50;
3. tasso di apprendimento best-matching = 0.1;
4. tasso di apprendimento finale = 0.01;
5. epoche di apprendimento per la connessione dei neuroni = 240;
6. tasso di decadimento del contatore di segnali ricevuti da un neurone = 0.05.

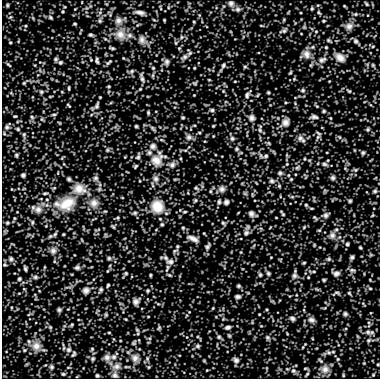
Risultati dell'apprendimento



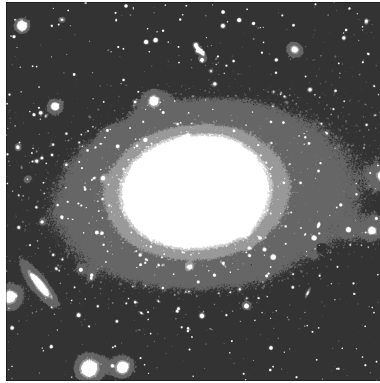
Ricordiamo dal capitolo 2 che, in fase di apprendimento la GCS non aggiorna tutti i pattern di riferimento. Tuttavia, sebbene durante l'effettivo funzionamento una parte dello spazio degli input non sarà adeguatamente rappresentato, la GCS risulta essere sufficientemente affidabile e veloce

3 Esperimenti di segmentazione

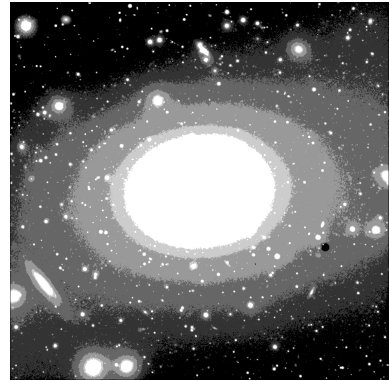
come primo (o, anche secondo, e in alcuni casi terzo) livello per una MultiLayerClusteringNN.



*Segmentazione 28:
MultiLayerClusteringNN
con 3 livelli di GCS (a 50,
20 e 6 neuroni)*



*Segmentazione 29:
MultiLayerClusteringNN
con 3 livelli di GCS (a 50,
20 e 6 neuroni)*



*Segmentazione 30:
MultiLayerClusteringNN
con 2 livelli di GCS (a 50 e
20 neuroni) e un livello
Neural Gas (a 6 neuroni)*

4 Conclusioni e sviluppi futuri

Il presente lavoro è stato mirato alla progettazione ed allo sviluppo di NExt2, un software per la segmentazione delle immagini astronomiche che risponde ad alcuni requisiti generali, che possono essere sintetizzati come segue:

- Configurabilità: particolare attenzione è stata destinata alla granularità della configurazione di ciascun componente. Questo obiettivo risulta complicato dalla necessità di conciliare l'uso del polimorfismo con le differenze degli algoritmi che implementano i diversi componenti (si consideri ad esempio la rete non supervisionata multilivello discussa nel dettaglio nel Cap.2).
- Modularità: l'adozione del paradigma di programmazione ad oggetti (OOP) consente di riutilizzare gli algoritmi neurali sviluppati anche indipendentemente dalla segmentazione delle immagini astronomiche. Il risultato del presente lavoro è una libreria condivisa C++ da includere fra i modelli di Data Mining del progetto DaME.
- Flessibilità: il polimorfismo e l'ereditarietà consentono da sole di fornire estrema flessibilità al codice. Tuttavia, per gli scopi del presente lavoro è stato necessario sviluppare un sistema di ripristino degli esperimenti che consentisse all'utente di ripristinare gli esperimenti da specifici punti di ripristino. In tal modo l'utente può salvare i risultati intermedi, riutilizzarli in configurazioni diverse e verificarne i risultati. Questa caratteristica è determinante in fase di testing, oltre che per testare prototipi di nuovi algoritmi che potrebbero migliorare il software.
- Supporto per immagini multi-banda. Sebbene non sia stato possibile testare questa specifica caratteristica, il segmentatore è stato sviluppato per poter segmentare immagini riprese in bande fotometriche diverse.

Ciascun componente è stato testato in due modi:

- Test di unità: verifica del funzionamento del componente a se stante.
- Misura delle prestazioni: verifica del risultato della segmentazione in risposta ad una variazione dei parametri di configurazione dei componenti.

Sul secondo tipo di test è necessario sottolineare che in mancanza delle procedure di valutazione degli oggetti, deblending e ricostruzione degli aloni non è possibile estrarre un catalogo di oggetti dalla segmentazione: senza tale catalogo non è possibile misurare la completezza e la contaminazione del catalogo e determinare quantitativamente il miglior insieme di parametri per la segmentazione.

L'integrazione di NExt2 nel progetto DaME aggiunge a quest'ultimo non solo uno strumento consistente per la segmentazione delle immagini, ma fornisce anche un'ampia classe di algoritmi neurali non supervisionati per il clustering e la riduzione di dimensionalità.

Uno degli aspetti più rilevanti di NExt2 è la possibilità di segmentare immagini multi-banda: questa caratteristica è unica nell'ambito dei programmi per la segmentazione, consentendo di utilizzare l'informazione disponibile direttamente nello spazio dei pixel anziché confrontando i cataloghi finali. Sebbene non sia stato possibile, nell'ambito del presente lavoro, testare e valutare le prestazioni di NExt2 nella segmentazione di immagini multi-banda, determinando il limite superiore (in termini di magnitudine apparente) alla rivelabilità degli oggetti.

Appendice

Casi d'uso

Caso d'Uso 1	RichiedeSegmentazioneAutomatica
Finalità nel Contesto	Il Client richiede a NExt2IS di segmentare un'immagine in modo completamente automatico, cioè senza dover specificare una sequenza di finestre significative, per l'addestramento delle FeaturesExtractionNN e MultiLayerClusteringNN.
Ambito	NExt2IS
Livello	Task-Primario
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione: nessun componente principale configurato; 2.in configurazione: almeno un componente principale configurato; 3.in attesa di richieste per la segmentazione automatica di un'immagine: tutti i componenti principali configurati; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine: solo FeaturesExtractionNN e MultiLayerClusteringNN, configurati.
Condizione finale di successo	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.
Condizione finale di insuccesso	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione
	NExt2IS ha sollevato l'eccezione appropriata e interrotto l'elaborazione in corso.
Attore Primario	Client (un' applicazione, o parte di essa).
Trigger	Il Client <u>ConfiguraMeaningFullWindowsExtractor(caso d'uso 4)</u>

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
Scenario Principale di successo	Passo	Client	NExt2IS
	1	<u>ConfiguraMeaningFullWindowsExtractor(caso d'uso 4)</u>	
	2	<u>ConfiguraFeaturesExtractionNN(caso d'uso 5)</u>	
	3	<u>ConfiguraMultiLayerClusteringNN(caso d'uso 6)</u>	
	4	Specifica l'immagine da segmentare	
	5	Seleziona la segmentazione automatica dell'immagine specificata	
	6		Estrae le finestre significative dall'immagine di input
	7		Addestra la FeaturesExtractionNN su Pattern estratti da tutte le finestre
	8		Addestra la MultiLayerClusteringNN sulle features rappresentanti i Pattern estratti
	9		Partiziona l'insieme di tutti i Pattern estratti dall'immagine in un numero di sottoinsiemi (cluster) pari al numero di classi di Pixel desiderato
	10		Classifica i cluster in base alla luminosità media dei Pixel, corrispondenti ai Pattern che li compongono.
	11		Fornisce la <i>maschera di segmentazione</i> dell'immagine di input
	12		Si pone in attesa di richieste di segmentazione automatica

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
VARIANTE 1.A	RichiedeSegmentazioneAutomaticaSenzaConfigurareMeaningFullWindowsExtractor		
Precondizioni	Il MeaningFullWindowsExtractor è stato configurato in una precedente richiesta per la segmentazione automatica di un'immagine		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Ritorna al passo 2 dello scenario principale di successo	
VARIANTE 2.A	RichiedeSegmentazioneAutomaticaSenzaConfigurareFeaturesExtractionNN		
Precondizioni	La FeaturesExtractionNN è stata configurata in una precedente richiesta per la segmentazione (semi-)automatica di un'immagine		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Ritorna al passo 3 dello scenario principale di successo	
VARIANTE 3.A	RichiedeSegmentazioneAutomaticaSenzaConfigurareMultiLayerClusteringNN		
Precondizioni	La MultiLayerClusteringNN è stata configurata in una precedente richiesta per la segmentazione (semi-)automatica di un'immagine		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input		

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Ritorna al passo 4 dello scenario principale di successo	
VARIANTE 4.A	RichiedeSegmentazioneAutomaticaConSalvataggioDs9		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha salvato nel file specificato dal Client le finestre significative dell'immagine, in formato ds9.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica il nome del file, per il salvataggio, in formato ds9, delle finestre significative estratte dall'immagine	
	2.a	Le ultime quattro lettere del nome specificato per il file sono .reg	
		Passo	Client
		1	Predisporre il file .reg specificato per il salvataggio delle finestre significative, in formato ds9.
		2	Ritorna al passo 4 dello scenario principale di successo
	2.b	Le ultime quattro lettere del nome specificato per il file non sono .reg	
		Passo	Client
		1	Interrompe l'elaborazione in corso.
		2	Solleva un'appropriata eccezione

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
VARIANTE 4.B	RichiedeSegmentazioneAutomaticaConSalvataggioDs9ERisultatiIntermedi		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha salvato nella directory di output specificata: 1.il file, di nome ds9.reg, contenente le finestre significative estratte dall'immagine di input, in formato ds9; 2.i seguenti file di ripristino: (I)recoverySegmentationFromFeaturesExtractionNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento delle FeaturesExtractionNN fino alla fine; (II)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (III)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 3.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica il nome della directory di output per il salvataggio: 1.in formato ds9 delle finestre significative estratte dall'immagine; 2.informazioni utili per il ripristino della segmentazione corrente, a partire da un punto sicuro (checkpoint); 3.il salvataggio, in formato .fits, della versione segmentata dell'immagine di input.	
	2		Predisporre il file ds9.reg per il salvataggio delle finestre significative, in formato leggibile da ds9.
	3	Ritorna al passo 4 dello scenario principale di successo	

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
VARIANTE 4.C	RichiedeSegmentazioneAutomaticaConRisultatiIntermedi		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha salvato nella directory di output specificata: 1.i seguenti file di ripristino: (I)recoverySegmentationFromFeaturesExtractionNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento delle FeaturesExtractionNN fino alla fine; (II)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (III)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica il nome della directory di output per salvare: 1.informazioni utili per il ripristino della segmentazione corrente, a partire da un punto sicuro (checkpoint); 2.il salvataggio, in formato .fits, della versione segmentata dell'immagine di input.	
	2	Ritorna al passo 4 dello scenario principale di successo	

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
ESTENZIONE 1-11.a	GestisceErroriSegmentazioneAutomatica		
Precondizione	NExt2IS ha rilevato un errore all'avvio, durante, o a fine della segmentazione.		
Postcondizione	NExt2IS ha interrotto la segmentazione.		
	NExt2IS ha sollevato l'eccezione appropriata.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe la segmentazione.
	2		Solleva un'appropriata eccezione.
ESTENSIONE 6-11.b	GestisceSalvataggioRisultatiIntermediSegmentazioneAutomatica		
Precondizione	Il Client ha richiesto il salvataggio dei risultati intermedi della corrente segmentazione.		
	NExt2IS ha estratto le finestre significative dall'immagine di input.		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha salvato nella directory di output specificata: 1. i seguenti file di ripristino: (I) recoverySegmentationFromFeaturesExtractionNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento delle FeaturesExtractionNN fino alla fine; (II) recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (III) recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2. il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la segmentazione.		

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica			
ESTENSIONE 6-11.b	GestisceSalvataggioRisultatiIntermediSegmentazioneAutomatica			
Scenario Alternativo	Passo	Client		NExt2IS
	1.a	NExt2IS ha estratto solo le finestre significative:		
		Passo	Client	NExt2IS
		1		Salva nel file recoverySegmentationFromFeaturesExtractionNNTraining le seguenti informazioni: 1.gli attuali dati di configurazione per la segmentazione automatica; 2.le finestre significative estratte.
		2		Ritorna al passo 7 dello scenario principale di successo
	1.b	NExt2IS ha addestrato la FeaturesExtractionNN:		
		Passo	Client	NExt2IS
		1		Salva nel file recoverySegmentationFromMultiLayerClusteringNNTraining le seguenti informazioni: 3.gli attuali dati di configurazione per la segmentazione automatica; 4.le finestre significative estratte; 5.i risultati dell'addestramento della FeaturesExtractionNN.
		2		Ritorna al passo 8 dello scenario principale di successo.
	1.c	NExt2IS ha addestrato la MultiLayerClusteringNN:		
		Passo	Client	NExt2IS
		1		Salva nel file recoverySegmentationFromMultiLayerClusteringNNTraining le seguenti informazioni: 6.gli attuali dati di configurazione per la segmentazione automatica; 7.le finestre significative estratte; 8.i risultati dell'addestramento delle FeaturesExtractionNN e MultiLayerClusteringNN.
		2		Ritorna al passo 8 dello scenario principale di successo.
	1.d	NExt2IS ha partizionato l'immagine di input:		
		Passo	Client	NExt2IS
		1		Salva nel file segmentedImage.fits, la versione segmentata dell'immagine di input.
		2		Ritorna al passo 12 dello scenario principale di successo.

Appendice

Caso d'Uso 1	RichiedeSegmentazioneAutomatica		
ESTENZIONE 7.c	GestisceSalvataggioDs9		
Precondizione	NExt2IS ha estratto le finestre significative dall'immagine di input.		
	Il Client ha richiesto il salvataggio delle finestre estratte, in formato ds9.		
Postcondizione	NExt2IS ha salvato le finestre significative in formato ds9, nel file predisposto.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input.		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione.		
	NExt2IS ha sollevato l'eccezione appropriata, interrotto la segmentazione.		
Scenario Alternativo	Passo	Client	NExt2IS
	1		Salva, nel file predisposto, le finestre estratte, attenendosi al seguente formato ds9: image;point(x1,y1) image;point(x2,y2) dove: 1.(x1,y1) sono le coordinate (riga, colonna) del primo Pixel nell'angolo superiore sinistro della finestra; 2.(x2,y2) sono le coordinate (riga, colonna) dell'ultimo Pixel nell'angolo inferiore destro della finestra.
	2		Ritorna al passo 7 dello scenario principale di successo

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica
Finalità nel Contesto	Il Client richiede a NExt2IS di segmentare un'immagine in modo semi-automatico, specificando una sequenza di finestre significative, per l'addestramento delle FeaturesExtractionNN e MultiLayerClusteringNN.
Ambito	NExt2IS
Livello	Task-Primario
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.
Condizione finale di successo	NExt2IS ha segmentato semi-automaticamente l'immagine di input, in base ai parametri di configurazione, e alle finestre significative, specificati dal Client.
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.
Condizione finale di insuccesso	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.
Attore Primario	Client (un' applicazione, o parte di essa).
Trigger	Il Client <u>ConfiguraFeaturesExtractionNN(caso d'uso 5)</u>

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica		
Scenario principale di successo	Passo	Client	NExt2IS
	1	<u>ConfiguraFeaturesExtractionNN(caso d'uso 5)</u>	
	2	<u>ConfiguraMultiLayerClusteringNN(caso d'uso 6)</u>	
	3	Specifica l'immagine da segmentare	
	4	Seleziona la segmentazione semi-automatica dell'immagine specificata, specificando le finestre significative da usare	
	5		Addestra la FeaturesExtractionNN su Pattern estratti da tutte le finestre
	6		Addestra la MultiLayerClusteringNN sulle features rappresentanti i Pattern estratti
	7		Partiziona l'insieme di tutti i Pattern estratti dall'immagine in un numero di sottoinsiemi (cluster) pari al numero di classi di Pixel desiderato
	8		Classifica i cluster in base alla luminosità media dei Pixel, corrispondenti ai Pattern che li compongono.
	9		Fornisce la <i>maschera di segmentazione</i> dell'immagine di input
	10		Si pone in attesa di richieste di segmentazione automatica

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica		
VARIANTE 1.A	RichiedeSegmentazioneSemiAutomaticaSenzaConfigurareFeaturesExtractionNN		
Precondizioni	La FeaturesExtractionNN è stata configurata in una precedente richiesta per la segmentazione (semi-)automatica di un'immagine		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Ritorna al passo 2 dello scenario principale di successo	
VARIANTE 2.A	RichiedeSegmentazioneSemiAutomaticaSenzaConfigurareMultiLayerClusteringNN		
Precondizioni	La MultiLayerClusteringNN è stata configurata in una precedente richiesta per la segmentazione (semi-)automatica di un'immagine		
Postcondizione 1	NExt2IS ha segmentato semi-automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Ritorna al passo 3 dello scenario principale di successo	

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica		
VARIANTE 3.A	RichiedeSegmentazioneSemiAutomaticaConRisultatiIntermedi		
Postcondizione 1	NExt2IS ha segmentato automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.		
	NExt2IS ha salvato nella directory di output specificata: 1.i seguenti file di ripristino: (I)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (II)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica il nome della directory di output per salvare: 1.informazioni utili per il ripristino della segmentazione corrente, a partire da un punto sicuro (checkpoint); 2.il salvataggio, in formato .fits, della versione segmentata dell'immagine di input.	
	2	Ritorna al passo 3 dello scenario principale di successo	
ESTENZIONE 1-9.a	GestisceErroriSegmentazioneSemiAutomatica		
Precondizione	NExt2IS ha rilevato un errore all'avvio, durante, o a fine della segmentazione.		
Postcondizione	NExt2IS ha interrotto la segmentazione.		
	NExt2IS ha sollevato l'eccezione appropriata.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe la segmentazione.
	2		Solleva un'appropriata eccezione.

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica
ESTENSIONE 6-9.b	GestisceSalvataggioRisultatiIntermediSegmentazioneSemiAutomatica
Precondizione	Il Client ha richiesto il salvataggio dei risultati intermedi della corrente segmentazione.
	NExt2IS ha addestrato la FeaturesExtractionNN.
Postcondizione 1	NExt2IS ha segmentato semi-automaticamente l'immagine di input, in base ai parametri di configurazione indicati dal Client.
	NExt2IS ha salvato nella directory di output specificata: 1.i seguenti file di ripristino: (I)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (II)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la segmentazione.

Appendice

Caso d'Uso 2	RichiedeSegmentazioneSemiAutomatica		
ESTENSIONE 6-9.b	GestisceSalvataggioRisultatiIntermediSegmentazioneSemiAutomatica		
Scenario Alternativo	Passo	Client	NExt2IS
	1.a	NExt2IS ha addestrato la FeaturesExtractionNN	
		Passo	Client
		1	Salva nel file recoverySegmentationFromMultiLayerClusteringNNTraining le seguenti informazioni: 1.gli attuali dati di configurazione per la segmentazione semi-automatica; 2.le finestre significative estratte; 3.i risultati dell'addestramento della FeaturesExtractionNN.
		2	Ritorna al passo 6 dello scenario principale di successo.
	1.b	NExt2IS ha addestrato la MultiLayerClusteringNN	
		Passo	Client
		1	Salva nel file recoverySegmentationFromMultiLayerClusteringNNTraining le seguenti informazioni: 4.gli attuali dati di configurazione per la segmentazione semi-automatica; 5.le finestre significative estratte; 6.i risultati dell'addestramento delle FeaturesExtractionNN e MultiLayerClusteringNN.
		2	Ritorna al passo 7 dello scenario principale di successo.
	1.c	NExt2IS ha partizionato l'immagine di input:	
		Passo	Client
		1	Salva nel file segmentedImage.fits, la versione segmentata dell'immagine di input.
		2	Ritorna al passo 10 dello scenario principale di successo.

Appendice

Caso d'Uso 4	ConfiguraMeaningFullWindowsExtractor		
Finalità nel Contesto	Il Client configura l'estrattore delle finestre significative (MeaningFullWindowsExtractor) per poter richiedere a NExt2IS di segmentare un'immagine in modo automatico.		
Ambito	NExt2IS		
Livello	Task-Primario		
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.		
Condizione finale di successo	NExt2IS ha configurato il MeaningFullWindowsExtractor, come specificato dal Client		
	NExt2IS si trova in uno dei seguenti stati: 1.in configurazione; 2.in attesa di richieste per la segmentazione (semi-)automatica di un'immagine.		
Condizione finale di insuccesso	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Attore Primario	Client (un' applicazione, o parte di essa).		
Trigger	Il Client specifica: 1.il numero di finestre significative da estrarre; 2.la lunghezza del lato di una finestra significativa; 3.il tipo di MeaningFullWindowsExtractor da usare; 4.una lista di parametri di configurazione specifici per il tipo di MeaningFullWindowsExtractor scelto.		
Scenario principale di successo	Passo	Client	NExt2IS
	1	Specifica: 1.il numero di finestre significative da estrarre; 2.la lunghezza del lato di una finestra significativa; 3.il tipo di MeaningFullWindowsExtractor da usare; 4.una lista di parametri specifici per il tipo di MeaningFullWindowsExtractor scelto.	
	2		Configura il MeaningFullWindowsExtractor in base a quanto specificato dal Client.

Appendice

Caso d'Uso 4	ConfiguraMeaningFullWindowsExtractor		
VARIANTE 1.A	ConfiguraMeaningFullWindowsExtractorDaFile		
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.		
Postcondizione 1	NExt2IS ha configurato il MeaningFullWindowsExtractor, come specificato dal Client		
	NExt2IS si trova in uno dei seguenti stati: 1.in configurazione; 2.in attesa di richieste per la segmentazione (semi-)automatica di un'immagine.		
Postcondizione 2	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica un file di configurazione	
	2		Ritorna al passo 2 dello scenario principale di successo
ESTENSIONE 2.a	GestisciErroriConfigurazioneMeaningFullWindowsExtractor		
Precondizione	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client.		
Postcondizione	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe la configurazione in corso.
	2		Solleva l'eccezione appropriata.

Appendice

Caso d'Uso 6	ConfiguraFeaturesExtractionNN		
Finalità nel Contesto	Il Client configura la rete neurale per la riduzione della dimensionalità dei Pattern (FeaturesExtractionNN) per poter richiedere a NExt2IS di segmentare un'immagine in modo (semi-)automatico.		
Ambito	NExt2IS		
Livello	Task-Primario		
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.		
Condizione finale di successo	NExt2IS ha configurato la FeaturesExtractionNN, come specificato dal Client.		
	NExt2IS si trova in uno dei seguenti stati: 1.in configurazione; 2.in attesa di richieste per la segmentazione semi-automatica di un'immagine. 3.in attesa di richieste per la segmentazione automatica di un'immagine.		
Condizione finale di insuccesso	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Attore Primario	Client (un' applicazione, o parte di essa).		
Trigger	Il Client specifica: 1.il numero di bande, che dovranno avere le immagini per essere segmentate; 2.il diametro dell'intorno di un Pixel, per estrarne il corrispondente Pattern; 3.il numero di caratteristiche (features) da estrarre per ciascun Pattern; 4.il tipo di FeaturesExtractionNN da usare; 5.una lista di parametri di configurazione specifici per il tipo di FeaturesExtractionNN scelto.		
Scenario principale di successo	Passo	Client	NExt2IS
	1	Specifica: 1.il numero di bande, che dovranno avere le immagini per essere segmentate; 2.il diametro dell'intorno di un Pixel, per estrarne il corrispondente Pattern; 3.il numero di caratteristiche (features) da estrarre per ciascun Pattern; 4.il tipo di FeaturesExtractionNN da usare; 5.una lista di parametri di configurazione specifici per il tipo di FeaturesExtractionNN scelto.	
	2		Configura la FeaturesExtractionNN in base a quanto specificato dal Client.

Appendice

Caso d'Uso 6	ConfiguraFeaturesExtractionNN		
VARIANTE 1.A	ConfiguraFeaturesExtractionNNDaFile		
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.		
Postcondizione 1	NExt2IS ha configurato la FeaturesExtractionNN, come specificato dal Client		
	NExt2IS si trova in uno dei seguenti stati: 1.in configurazione; 2.in attesa di richieste per la segmentazione semi-automatica di un'immagine; 3.in attesa di richieste per la segmentazione automatica di un'immagine.		
Postcondizione 2	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica un file di configurazione	
	2		Ritorna al passo 2 dello scenario principale di successo
ESTENSIONE 2.a	GestisciErroriConfigurazioneFeaturesExtractionNN		
Precondizione	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client.		
Postcondizione	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe la configurazione in corso.
	2		Solleva l'eccezione appropriata.

Appendice

Caso d'Uso 7	ConfiguraMultiLayerClusteringNN		
VARIANTE 1.A	ConfiguraMultiLayerClusteringNNDaFile		
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.		
Postcondizione 1	NExt2IS ha configurato la MultiLayerClusteringNN, come specificato dal Client		
	NExt2IS si trova in uno dei seguenti stati: 1.in configurazione; 2.in attesa di richieste per la segmentazione semi-automatica di un'immagine; 3.in attesa di richieste per la segmentazione automatica di un'immagine.		
Postcondizione 2	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
		Specifica un file di configurazione	
			Ritorna al passo 2 dello scenario principale di successo
ESTENSIONE 2.a	GestisciErroriConfigurazioneFeaturesExtractionNN		
Precondizione	NExt2IS ha rilevato un errore nelle specifiche di configurazione fornite dal Client.		
Postcondizione	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la configurazione in corso.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe la configurazione in corso.
	2		Solleva l'eccezione appropriata.

Appendice

Caso d'Uso 3	RipristinaSegmentazione
Finalità nel Contesto	Il Client richiede a NExt2IS di ripristinare la segmentazione di un'immagine riavviandola da un punto sicuro (checkpoint). Le informazioni di ripristino relative a ciascun checkpoint sono contenute in uno dei file usati per il salvataggio dei risultati intermedi. I checkpoint di segmentazione sono tre: 1.il ripristino dal primo checkpoint avviene in base alle informazioni contenute nel file recoverySegmentationFromFeaturesExtractionNNTraining; 2.il ripristino dal secondo checkpoint avviene in base alle informazioni contenute nel file recoverySegmentationFromMultiLayerClusteringNNTraining; 3.il ripristino dal terzo checkpoint avviene in base alle informazioni contenute nel file recoverySegmentationFromPixelsClustersIndividuation. Il ripristino può avvenire anche da file con un nome diverso, ma che contengano le informazioni di uno dei file sopraelencati.
Ambito	NExt2IS
Livello	Task-Primario
Precondizione	NExt2IS si trova in uno dei seguenti stati: 1.attesa di configurazione; 2.in configurazione; 3.in attesa di richieste per la segmentazione automatica di un'immagine; 4.in attesa di richieste per la segmentazione semi-automatica di un'immagine.
Condizione finale di successo	NExt2IS ha completato la segmentazione dell'immagine di input, in base alle informazioni contenute nel file di ripristino, specificato dal Client.
	NExt2IS ha fornito al Client la <i>maschera di segmentazione</i> dell'immagine di input
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.
Condizione finale di insuccesso	NExt2IS ha rilevato un errore all'avvio, durante, o a fine del ripristino della segmentazione, o del suo completamento.
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.
Attore Primario	Client (un' applicazione, o parte di essa).
Trigger	Il Client specifica il file per il ripristino della segmentazione

Appendice

Caso d'Uso 3	RipristinaSegmentazione		
Scenario principale di successo	Passo	Client	NEx2IS
	1	Seleziona il file per il ripristino della segmentazione.	
	2	Specifica l'immagine di input la cui segmentazione desidera ripristinare.	
	3	Seleziona il ripristino della segmentazione dell'immagine di input	
	4		Ripristina i suoi componenti principali in base alle informazioni contenute nel file di input.
	5		Addestra la FeaturesExtractionNN su Pattern estratti da tutte le finestre.
	6		Addestra la MultiLayerClusteringNN sulle features rappresentanti i Pattern estratti
	7		Partiziona l'insieme di tutti i Pattern estratti dall'immagine in un numero di sottoinsiemi (cluster) pari al numero di classi di Pixel desiderato
	8		Classifica i cluster in base alla luminosità media dei Pixel, corrispondenti ai Pattern che li compongono.
	9		Costruisce la <i>maschera di segmentazione</i> dell'immagine di input
	10		Si pone in attesa di richieste di segmentazione automatica

Appendice

Caso d'Uso 3	RipristinaSegmentazione		
VARIANTE 3.A	RipristinaSegmentazioneConSalvataggioRisultatiIntermedi		
Postcondizione 1	NExt2IS ha completato la segmentazione dell'immagine di input, in base alle informazioni contenute nel file di ripristino, specificato dal Client.		
	NExt2IS ha salvato nella directory di output specificata: 1.i seguenti file di ripristino, il primo nel caso la segmentazione riprenda dal primo checkpoint, il secondo nel caso la segmentazione riprenda dal primo o secondo checkpoint: (I)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (II)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario Alternativo	Passo	Client	NExt2IS
	1	Specifica il nome della directory di output per salvare: 1.informazioni utili per il ripristino della segmentazione corrente, a partire da un punto sicuro (checkpoint); 2.il salvataggio, in formato .fits, della versione segmentata dell'immagine di input.	
	2	Ritorna al passo 3 dello scenario principale di successo	
ESTENSIONE 1-10.a	GestisceErroriRipristinoSegmentazione		
Precondizione	NExt2IS ha rilevato un errore all'avvio, durante, o a fine del ripristino della segmentazione, o del suo completamento.		
Postcondizione	NExt2IS ha interrotto l'elaborazione in corso.		
	NExt2IS ha sollevato l'eccezione appropriata.		
Scenario di Errore	Passo	Client	NExt2IS
	1		Interrompe l'elaborazione in corso.
	2		Solleva un'appropriata eccezione.

Appendice

Caso d'Uso 3	RipristinaSegmentazione		
ESTENZIONE 4.b	RipristinaSegmentazioneDaSecondoCheckpoint		
Precondizione	Il file di ripristino è compatibile con l'immagine di input		
	Il file di ripristino contiene le informazioni per riprendere la segmentazione dell'immagine di input dal secondo checkpoint		
Postcondizione 1	NExt2IS ha completato la segmentazione dell'immagine di input, in base alle informazioni contenute nel file di ripristino, specificato dal Client.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante, o a fine del ripristino della segmentazione, o del suo completamento.		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario alternativo	Passo	Client	NExt2IS
	1		Ritorna al passo 6 dello scenario principale di successo
ESTENZIONE 4.c	RipristinaSegmentazioneDaTerzoCheckpoint		
Precondizione	Il file di ripristino è compatibile con l'immagine di input		
	Il file di ripristino contiene le informazioni per riprendere la segmentazione dell'immagine di input dal terzo checkpoint		
Postcondizione 1	NExt2IS ha completato la segmentazione dell'immagine di input, in base alle informazioni contenute nel file di ripristino, specificato dal Client.		
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input		
	NExt2IS è in attesa di richieste per la segmentazione automatica.		
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante, o a fine del ripristino della segmentazione, o del suo completamento.		
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto l'elaborazione in corso.		
Scenario alternativo	Passo	Client	NExt2IS
	1		Ritorna al passo 7 dello scenario principale di successo

Appendice

Caso d'Uso 3	RipristinaSegmentazione
ESTENSIONE 6-9.b	GestisceSalvataggioRisultatiIntermediRipristinoSegmentazione
Precondizione	Il file di ripristino è compatibile con l'immagine di input
	Il file di ripristino contiene le informazioni per riprendere la segmentazione dell'immagine di input da un checkpoint valido
	Il Client ha richiesto il salvataggio dei risultati intermedi.
Postcondizione 1	NExt2IS ha completato la segmentazione dell'immagine di input, in base alle informazioni contenute nel file di ripristino, specificato dal Client.
	NExt2IS ha salvato nella directory di output specificata: 1.i seguenti file di ripristino, il primo nel caso la segmentazione riprenda dal primo checkpoint, il secondo nel caso la segmentazione riprenda dal primo o secondo checkpoint: (I)recoverySegmentationFromMultiLayerClusteringNNTraining, contenente le informazioni per ripristinare la segmentazione dall'addestramento della MultiLayerClusteringNN fino alla fine; (II)recoverySegmentationFromPixelsClustersIndividuation, contenente le informazioni per ripristinare la segmentazione dall'individuazione dei cluster di appartenenza dei pixel dell'immagine, fino alla fine; 2.il file, di nome segmentedImage.fits, contenente la versione segmentata dell'immagine di input.
	NExt2IS ha costruito la <i>maschera di segmentazione</i> dell'immagine di input
	NExt2IS è in attesa di richieste per la segmentazione (semi-)automatica.
Postcondizione 2	NExt2IS ha rilevato un errore all'avvio, durante o alla fine della segmentazione
	NExt2IS ha sollevato l'eccezione appropriata, e interrotto la segmentazione.

Appendice

Caso d'Uso 3	RichiedeRipristinoSegmentazione			
ESTENSIONE 6-9.b	GestisceSalvataggioRisultatiIntermediRipristinoSegmentazione			
Scenario Alternativo	Passo	Client		NExt2IS
	1.a	NExt2IS ha ripreso la segmentazione dell'immagine di input dal primo checkpoint ed ha addestrato la FeaturesExtractionNN		
		Passo	Client	NExt2IS
		1		Salva nel file recoverySegmentationFromMultiLayerClusteringNNTraining le seguenti informazioni: 1.gli attuali dati di configurazione per la segmentazione semi-automatica; 2.le finestre significative estratte; 3.i risultati dell'addestramento della FeaturesExtractionNN.
		2		Ritorna al passo 6 dello scenario principale di successo.
	1.b	NExt2IS ha ripreso la segmentazione dell'immagine di input dal secondo checkpoint ed ha addestrato la MultiLayerClusteringNN		
		Passo	Client	NExt2IS
		1		Salva nel file recoverySegmentationFromPixelsClustersIndividuation le seguenti informazioni: 4.gli attuali dati di configurazione per la segmentazione semi-automatica; 5.le finestre significative estratte; 6.i risultati dell'addestramento delle FeaturesExtractionNN e MultiLayerClusteringNN.
		2		Ritorna al passo 7 dello scenario principale di successo.
	1.c	NExt2IS ha ripreso la segmentazione da un checkpoint valido ed ha partizionato l'immagine di input:		
		Passo	Client	NExt2IS
		1		Salva nel file segmentedImage.fits, la versione segmentata dell'immagine di input.
		2		Ritorna al passo 10 dello scenario principale di successo.

Esempio di file di configurazione per il MeaningFullWindowsExtractor

```
begin
    meaningFullWindowsNumber 10
    windowSideLength 25
    meaningFullWindowsExtractorType HistogramBasedMeaningFullWindowsExtractor
    specificParameter binsNumber 6
    specificParameter subImageSideLength 100
end
```

Esempio di file di configurazione per la FeaturesExtractionNN

```
begin
    bandsNumber 1
    pixelNeighborhoodDiameter 5
    featuresNumber 3
    featuresExtractionNNType StandardPCANN
    specificParameter learningRate 0.03
    specificParameter stopLearningThreshold 0.00001
    specificParameter learningFeedBackType SYMMETRICAL
end
```

Esempio di file di configurazione per la MultiLayerClusteringNN

```
layer bottom
begin
    layerClustersNumber 50
    layerClusteringNNType NeuralGasClusteringNN
    specificParameter initialLearningRate 0.7
    specificParameter finalLearningRate 0.05
    specificParameter learningEpochsNumber 50
    specificParameter initialVariance 10
    specificParameter finalVariance 0.1
end
layer middle
begin
    layerClustersNumber 20
    layerClusteringNNType NeuralGasClusteringNN
    specificParameter initialLearningRate 0.7
    specificParameter finalLearningRate 0.05
    specificParameter learningEpochsNumber 50
    specificParameter initialVariance 10
```

Appendice

```
specificParameter finalVariance 0.1
end
layer top
begin
    layerClustersNumber 6
    layerClusteringNNType NeuralGasClusteringNN
    specificParameter initialLearningRate 0.7
    specificParameter finalLearningRate 0.05
    specificParameter learningEpochsNumber 50
    specificParameter initialVariance 10
    specificParameter finalVariance 0.1
end
```

Esempio di file di configurazione per la segmentazione automatica

```
component_to_configure meaningFullWindowsExtractor
```

Si veda il paragrafo Esempio di file di configurazione per il MeaningFullWindowsExtractor.

```
component_to_configure featuresExtractionNN
```

Si veda il paragrafo Esempio di file di configurazione per la FeaturesExtractionNN.

```
component_to_configure multiLayerClusteringNN
```

Si veda il paragrafo Esempio di file di configurazione per la MultiLayerClusteringNN.

Esempio di file di configurazione per la segmentazione semi-automatica

```
component_to_configure featuresExtractionNN
```

Si veda il paragrafo Esempio di file di configurazione per la FeaturesExtractionNN.

```
component_to_configure multiLayerClusteringNN
```

Si veda il paragrafo Esempio di file di configurazione per la MultiLayerClusteringNN.

Esempio di file leggibile da ds9

Nome: ds9.reg

```
image;point (724,862)    ←  indicano le coordinate del primo Pixel della prima finestra usata
                           da NExt2IS
image;point (748,886)    ←  indicano le coordinate dell'ultimo Pixel della prima finestra usata
                           da NExt2IS
.
.
.
image;point (170,427)    ←  indicano le coordinate del primo Pixel dell'ultima finestra usata
                           da NExt2IS
image;point (194,451)    ←  indicano le coordinate dell'ultimo Pixel dell'ultima finestra usata
                           da NExt2IS
```

Esempio di file di ripristino

Nome: recoverySegmentationFromPixelsClustersIndividuation

```
1      1      0      0      1      1      0      0 ←  indicano che NExt2IS ha appena
                                                         terminato l'addestramento della
                                                         MultiLayerClusteringNN
HistogramBasedMeaningFullWindowsExtractor ←  indica il tipo di
                                                         MeaningFullWindowsExtractor usato
0      1      0      ←  indicano che l'HistogramBasedMeaningFullWindowsExtractor è
                                                         in attesa di richieste per l'estrazione di finestre significative
10     25     ←  dati di configurazione l'HistogramBasedMeaningFullWindowsExtractor:
                                                         il numero di finestre significative da estrarre, la lunghezza del lato di una
                                                         finestra,
6      100     ←  il numero di bin da usare, e la lunghezza del lato di una sotto-immagine
10     ←  indica il numero di finestre significative usate da NExt2IS
25     ←  indica la lunghezza del lato della prima finestra usata da NExt2IS
724    862    ←  indicano le coordinate del primo Pixel della prima finestra usata da NExt2IS
748    886    ←  indicano le coordinate dell'ultimo Pixel della prima finestra usata da NExt2IS
.
.
.
25     ←  indica la lunghezza del lato dell'ultima finestra usata da NExt2IS
170    427    ←  indicano le coordinate del primo Pixel dell'ultima finestra usata da NExt2IS
194    451    ←  indicano le coordinate dell'ultimo Pixel dell'ultima finestra usata da NExt2IS
```


Appendice

1 5 ← indicano il numero di bande e il diametro dell'intorno di un Pixel usati da NExt2IS

StandardPCANN ← indica il tipo di FeaturesExtractionNN usato da NExt2IS

0 0 0 1 ← indicano che la StandardPCANN è addestrata

25 3 ← indicano la lunghezza del Pattern di input e il numero di features, per cui la StandardPCANN è configurata

0.001 1e-05 HIERARCHICAL ← indicano il tasso di apprendimento, la soglia di arresto del livello di apprendimento e il tipo di feed-back di apprendimento, per la StandardPCANN è configurata. A seguire, sono quindi riportati, i dati relativi a ciascun Pattern di riferimento: la sua lunghezza e le sue componenti di ciascun Pattern di riferimento

25

0.0346509	0.175133	0.303468	0.0331769	0.130051	0.0606858	0.246448
0.0974224	0.145512	0.231024	0.203425	0.221366	0.142638	
0.0976295	0.0093447	0.312	0.285536	0.120127	0.220385	0.251954
0.266847	0.311088	0.258262	0.174911	0.136325		

25

0.085706	0.265486	0.0672361	0.159317	0.293613	0.229263	0.176988
0.149498	0.214504	0.20274	0.271599	0.265256	0.13566	
0.0491553	0.0976402	0.0414848	0.244629	0.315186	0.185118	
0.0193111	0.309166	0.179068	0.308474	0.109915	0.0882894	

25

0.257697	0.0514698	0.0778313	0.194331	0.231497	0.219394	0.264335
0.181156	0.289102	0.0902765	0.140685	0.183444	0.280675	
0.303345	0.0879981	0.156943	0.236704	0.0181172	0.291524	
0.292664	0.103248	0.00466474	0.205946	0.106697	0.207088	

0 0 0 0 1 ← indicano che la MultiLayerClusteringNN è addestrata

3 3 ← indicano il numero di livelli e la lunghezza del Pattern di input, per cui la MultiLayerClusteringNN è configurata

GCSClusteringNN ← indica il tipo della ClusteringNN usata al primo livello (bottom) della MultiLayerClusteringNN

0 0 0 1 ← indicano che la GCSClusteringNN è addestrata

3 30 ← indicano la lunghezza del Pattern di input e il numero di cluster della GCSClusteringNN

0.1 0.01 1000 ← indicano i tassi di apprendimento best-matching e neighborhood, e il numero di epoche per la connessione di un neurone della GCSClusteringNN

0.05 ← indicano il tasso di decadimento dei contatori dei segnali ricevuti da ciascun neurone della GCSClusteringNN.

Appendice

3				←	indica la lunghezza del primo Pattern di riferimento della GCSClusteringNN
117496	117315	117116		←	indicano le componenti del primo Pattern di riferimento della GCSClusteringNN
.					
.					
.					
3				←	indica la lunghezza dell'ultimo Pattern di riferimento della GCSClusteringNN
83750.3	83649	83929.1		←	indicano le componenti dell'ultimo Pattern di riferimento della GCSClusteringNN
4				←	indica il numero dei vicini del primo neurone (cluster) della GCSClusteringNN
7	4	3	2	←	indicano i vicini del primo neurone (cluster) della GCSClusteringNN
.					
.					
.					
4				←	indica il numero dei vicini dell'ultimo neurone (cluster) della GCSClusteringNN
12	16	23	27	←	indicano i vicini dell'ultimo neurone (cluster) della GCSClusteringNN
NeuralGasClusteringNN				←	indica il tipo della ClusteringNN usata al secondo livello (middle) della MultiLayerClusteringNN
0	0	0	1	←	indicano che la NeuralGasClusteringNN è addestrata
3	20			←	indicano la lunghezza del Pattern di input e il numero di cluster della NeuralGasClusteringNN
0.7	0.05	50		←	indicano i tassi di apprendimento iniziale e finale, e il numero di epoche di apprendimento della NeuralGasClusteringNN
3	0.1			←	indicano le varianze iniziale e finale della NeuralGasClusteringNN
3				←	indica la lunghezza del primo Pattern di riferimento della NeuralGasClusteringNN
91418.9	91695.3	91127.8		←	indicano le componenti del primo Pattern di riferimento della NeuralGasClusteringNN
.					
.					
.					

Appendice

3				←	indica la lunghezza dell'ultimo Pattern di riferimento della NeuralGasClusteringNN
150432	152914	153744		←	indicano le componenti dell'ultimo Pattern di riferimento della NeuralGasClusteringNN
NeuralGasClusteringNN				←	indica il tipo della ClusteringNN usata al terzo livello (top) della MultiLayerClusteringNN
0	0	0	1	←	indicano che la NeuralGasClusteringNN è addestrata
3	6			←	indicano la lunghezza del Pattern di input e il numero di cluster della NeuralGasClusteringNN
0.7	0.05	50		←	indicano i tassi di apprendimento iniziale e finale, e il numero di epoche di apprendimento della NeuralGasClusteringNN
2	0.1			←	indicano le varianze iniziale e finale della NeuralGasClusteringNN
3				←	indica la lunghezza del primo Pattern di riferimento della NeuralGasClusteringNN
83954.4	83852.4	84130.4		←	indicano le componenti del primo Pattern di riferimento della NeuralGasClusteringNN
.					
.					
.					
3				←	indica la lunghezza dell'ultimo Pattern di riferimento della NeuralGasClusteringNN
89107.7	89263.2	90061.1		←	indicano le componenti dell'ultimo Pattern di riferimento della NeuralGasClusteringNN

Ringraziamenti

Grazie alla mia famiglia per tutto.

Grazie ai proff. Isgrò e Longo per avermi dato la possibilità di lavorare nel settore dell'astro-informatica.

Grazie a Omar Laurino per avermi, insieme ai proff. Isgrò e Longo, seguito durante lo sviluppo e messa a punto di NExt2.

Grazie ai ragazzi/e del laboratorio di Astrofisica e di S.C.O.P.E per avermi reso più divertente il lavoro.

Grazie a Marisa, Fabrizio, Marco, Mara, Giovanna, Raffaele e Nadia, e ai ragazzi/e dell'auletta di Fisica per le ore di studio e relax passate assieme.

Grazie ai ragazzi/e della C4 perché NOI LA CRISI NON LA PAGHIAMO.