

Analisi di immagini mediche con il software libero



Niccolò Camarlinghi

niccolo.camarlinghi@df.unipi.it

GULP
Stazione Leopolda,
16 marzo 2011

Premessa



Questo talk è basato sulla mia esperienza personale, quindi contiene moltissimi punti di vista soggettivi !!

Obiettivi di questo seminario

Dare una panoramica di alcune problematiche presenti nel campo dell'analisi delle immagini mediche e soprattutto dei software open source disponibili al momento.

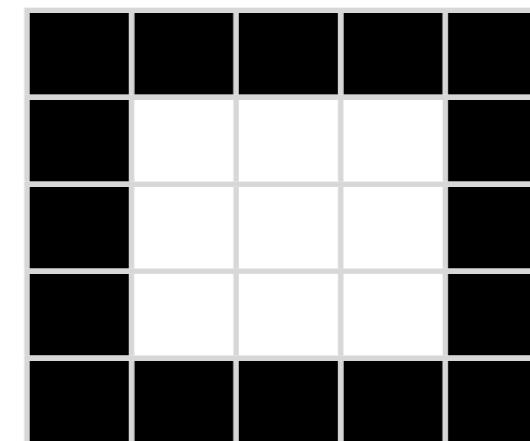
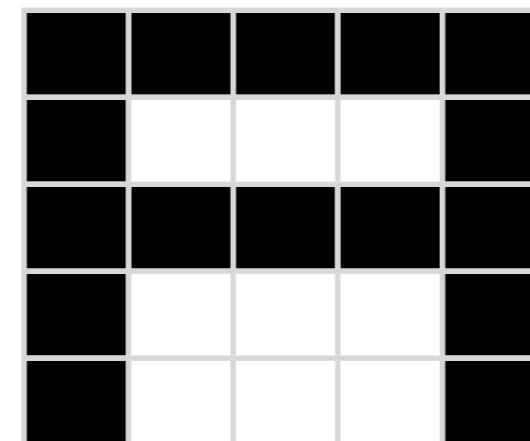
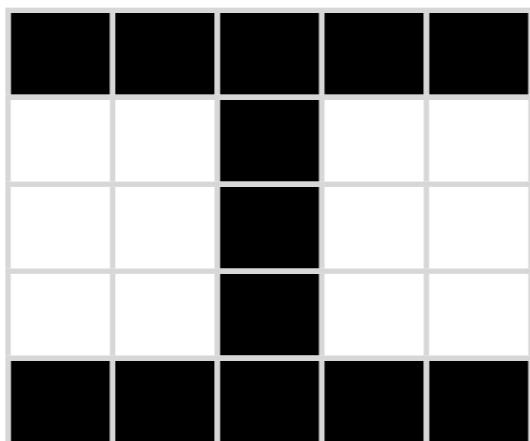
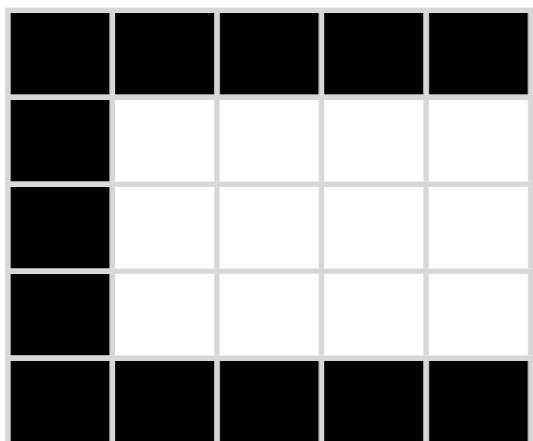
Outline

- **Parte I: Basi**
 - Analisi delle immagini digitali
 - Apprendimento automatico e classificatori
 - Sistemi automatici per la lettura di immagini mediche
- **Parte II: Software open source utili per l'analisi delle immagini mediche e non**
 - ITK
 - VTK
 - CMake
 - FANN
 - SVM_LIGHT
 - Osirix
- **Parte III: Esempio di applicazione sviluppata con software opensource**
 - CADe per l'individuazione di noduli polmonari in immagini TAC
- **Conclusioni**

Immagini digitali

Immagini digitali I / 2

- Rappresentabili tramite matrici
- Esempio di immagine B/N: ogni elemento di ciascuna matrice è trasformato in una sfumatura di grigio a seconda del suo valore



I	I	I	I	I
I	0	0	0	0
I	0	0	0	0
I	0	0	0	0
I	I	I	I	I

I	I	I	I	I
0	0	I	0	0
0	0	I	0	0
0	0	I	0	0
I	I	I	I	I

I	I	I	I	I
I	0	0	0	I
I	I	I	I	I
I	0	0	0	I
I	0	0	0	I

I	I	I	I	I
I	0	0	0	I
I	0	0	0	I
I	0	0	0	I
I	I	I	I	I

Immagini digitali 2/2

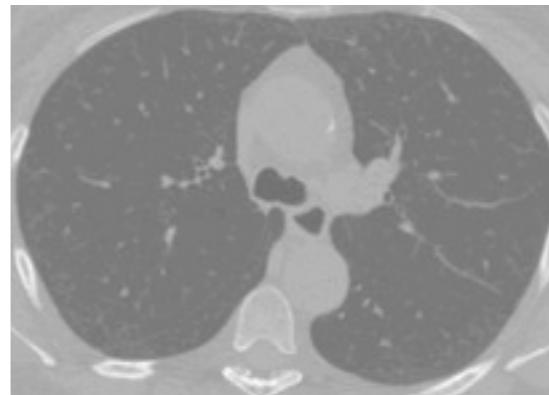
- L'elemento fondamentale di un'immagine digitale è il pixel (per immagini 3d si usa anche la dicitura voxel)
- Il pixel di un'immagine può contenere vari tipi di dati: interi, float o anche vettori di float nel caso di immagini RGB (Red Green Blue)
- Le immagini possono essere in linea di principio n-dimensional
- Tipicamente si utilizzano immagini di 2/3 dimensioni
- Esempio: Radiografia immagine 2D, TAC immagine 3D

Analisi delle immagini

- E' una disciplina il cui scopo è estrarre informazioni da immagini digitali tramite algoritmi
- E' un campo relativamente nuovo, multidisciplinare, in cui sono necessarie varie competenze: matematica, informatica,fisica e creatività

Un po' di terminologia

- Segmentazione: processo in cui si partiziona un'immagine in regioni significative, ad es. si individuano parti anatomiche di un paziente
- Filtro: funzione che prende in input un'immagine e produce una o più immagini in output
- Filtro gaussiano (smoothing): filtro che effettua una convoluzione tra una immagine e una gaussiana



Apprendimento automatico

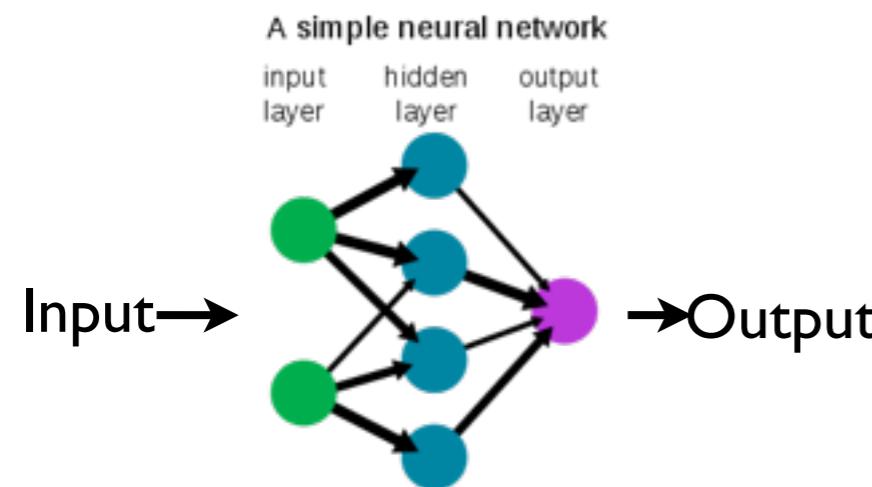
Cos'è l'apprendimento automatico?

- E' una forma di intelligenza artificiale
- Si basa su sintetizzare conoscenza attraverso esempi
- Gli approcci di cui parleremo sono essenzialmente reti neurali e support vector machine

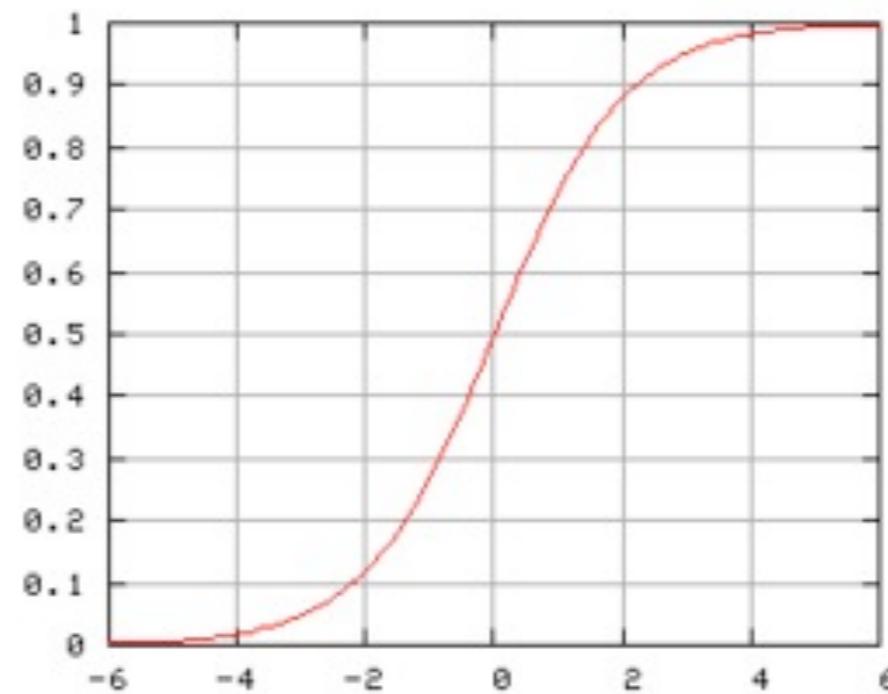
Reti neurali

- Sono modelli matematici che traggono ispirazione dalla biologia
- Sono composte da neuroni interconnessi tra loro, che formano un sistema non lineare
- Un tipo molto comune sono le reti “feed-forward” che sono composte da un numero variabile di neuroni distribuiti su più strati
- Possono essere usate per dividere i dati in classi (classificazione) o per interpolare una funzione (regressione)

Reti “Feed Forward” (FF)



$$N_j = \phi\left(\sum_{i \in \text{layer } j-1} W_{i,j} N_i\right)$$



N_j → Valore del j-esimo neurone

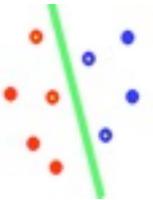
ϕ → Funzione di attivazione del neurone

$W_{i,j}$ → Sinapsi tra il neurone i e j

Tutto questo per dire...

- Che le reti FF sono un sistema complesso capace di immagazzinare informazioni nelle sinapsi $W_{i,j}$
- L'input delle reti sono dati che rappresentano il nostro modo di descrivere il fenomeno

Support Vector Machine (SVM)

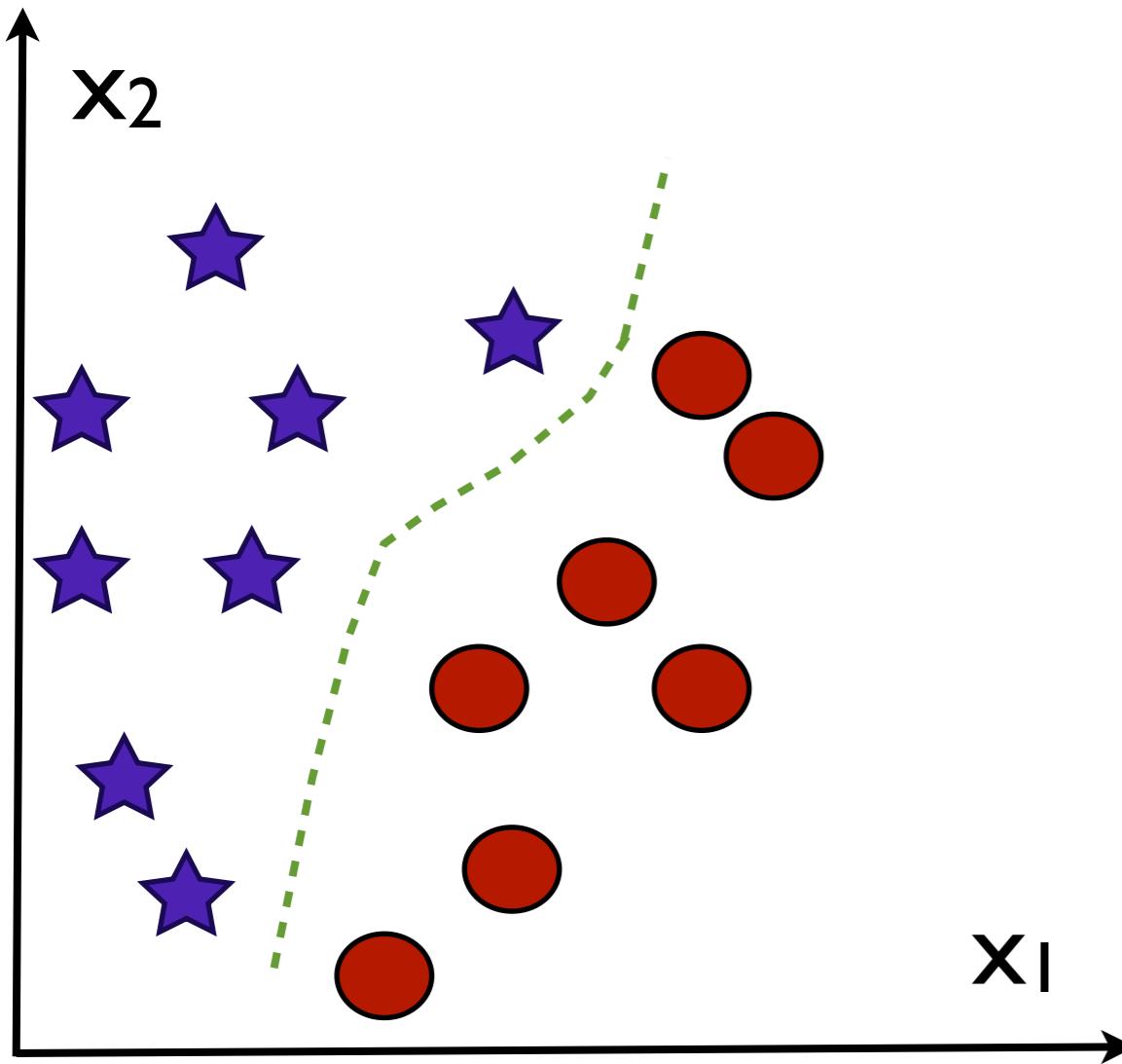


- Sono dei classificatori come le reti neurali
- Non si basano su un'analogia biologica ma sul trovare l'iper-piano che separa “meglio” i dati
- Sono modelli meglio definiti dal punto di vista matematico rispetto alle reti neurali

Esempio

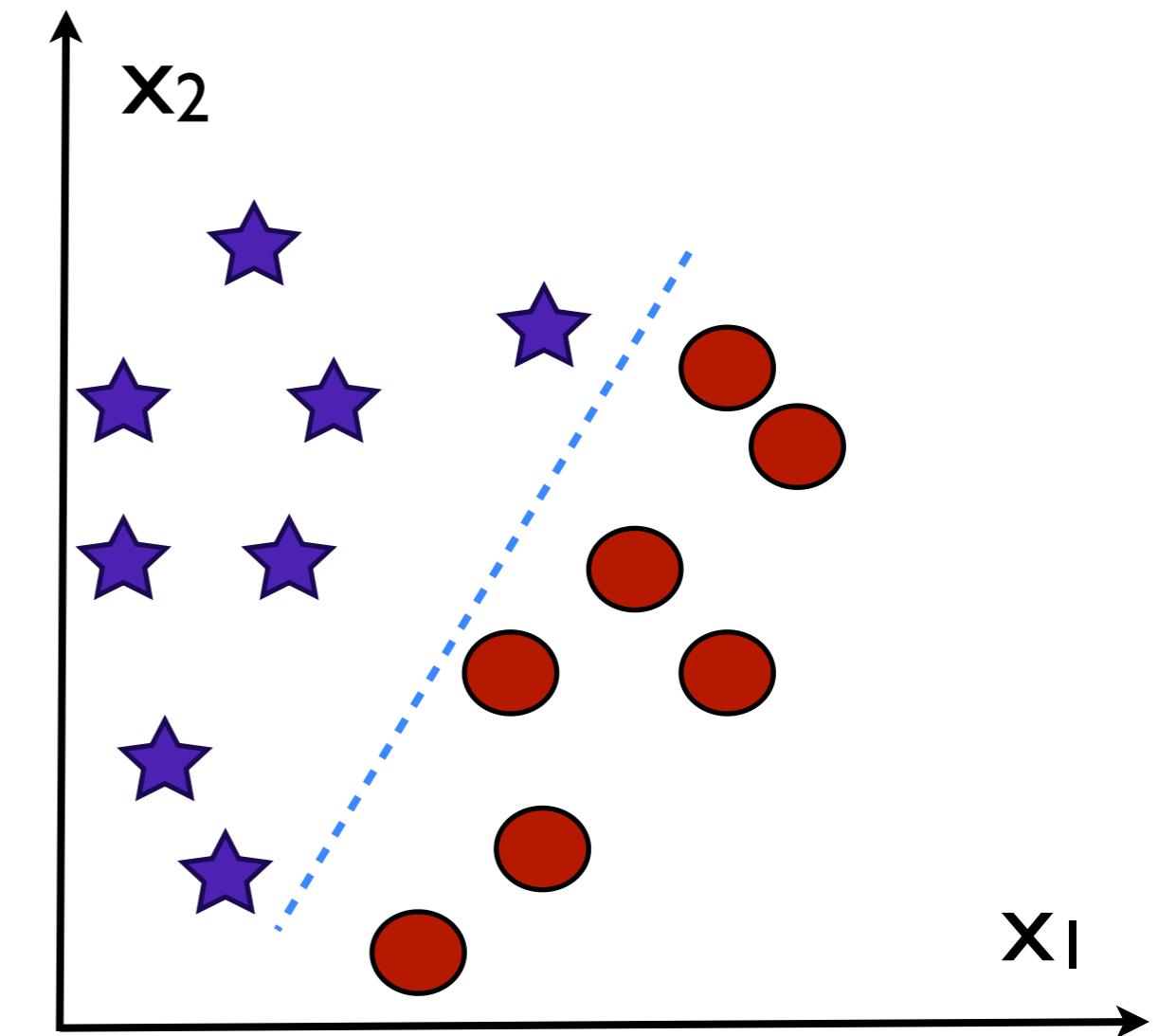
Rete neurale

--- Superficie separatrice



SVM

--- Iper-piano separatore



x_1, x_2 sono due variabili che descrivono le nostre osservazioni

Come si utilizzano i classificatori?

- La vita di un classificatore prevede due fasi: allenamento e utilizzo vero e proprio
- Nella fase di allenamento si cerca di far apprendere al classificatore il fenomeno che vogliamo modellizzare
- Nella fase di utilizzo sfruttiamo la conoscenza appresa nella fase di allenamento, per dare delle previsioni su dati mai visti prima

Quando si utilizzano?

- Si utilizzano per trattare problemi di difficile soluzione con algoritmi tradizionali
- Tipicamente in problemi in cui si conoscono i risultati senza saper dare un criterio
- Approccio naturale nel “mondo” delle immagini mediche in cui non ci sono forti fondamenti teorici
- Si cerca di estrarre la conoscenza dall’esperienza dei lettori esperti

Imaging medico e sistemi automatici

Cos'è l'imaging medico?

L'imaging medico è un insieme di tecniche usate per creare immagini del corpo umano o di parti di esso per motivi clinici, ad esempio diagnosi di patologie, o per monitorare l'andamento di una terapia

Esempi

- Radiografia o RX
- TAC (Tomografia Assiale Computerizzata)
aka CT (Computed Tomography)
- Risonanza magnetica (RM)
- PET (Positron Emission Tomography)

Esempio di RX torace



Nodulo nel lobo polmonare inferiore destro

Interpretazione e lettura di immagini mediche

- E' esperienza comune che le immagini mediche non siano facili da leggere per lettori non esperti (vedi slide precedente)
- Da qui nasce l'idea di sviluppare sistemi software per l'ausilio alla lettura delle immagini mediche

Si possono individuare vari sistemi automatici

Diagnosi: Computer
Aided Diagnosis (CADx)



Questo paziente ha un
tumore?

Individuazione di
patologie: Computer
Aided Detection (CADe)



Trova i noduli presenti in
questa immagine

Quantificazione di
patologie



Quanto è estesa la
polmonite di questo
paziente?

N.B. Il software di lettura delle immagini non è tipicamente
volto a sostituire il medico ma piuttosto ad affiancarlo nel suo
lavoro

Quando ha senso ricorrere a un sistema automatico?

- In linea di massima sempre (dal punto di vista scientifico)
- Nella pratica solo per questioni che sono problematiche anche per i medici/radiologi
- Soprattutto per i problemi di quantificazione, che richiedono stime accurate

Parte II

Software open source

Una nota filosofica: perché il software scientifico deve essere open source?

- Necessità di conoscere i dettagli dell'implementazione degli algoritmi
- Necessità di modificare il software
- Facilita il debug degli algoritmi
- Riproducibilità dei risultati con altri software

Open source applicato ai dati

- Sviluppare un software per l'ausilio alla lettura di immagini mediche richiede di possedere tante immagini annotate da lettori esperti
- Scientificamente parlando è necessario confrontare diversi software per stabilire la validità di ciascun approccio
- Per fare questo è necessario che tutti i gruppi di ricerca abbiano accesso a database pubblici di dati annotati (ad es. Lung Image Database Consortium LIDC)
- Solo così un lavoro può dirsi pienamente scientifico, cioè in grado di rispettare il criterio confrontabilità e riproducibilità

Cosa serve per costruire un buon sistema di ausilio alla lettura delle immagini mediche?

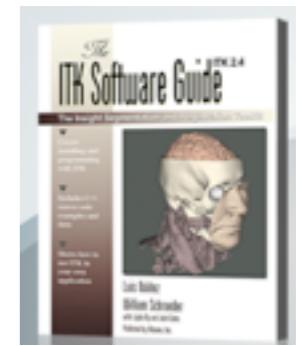
- Un framework per Analisi delle immagini
- Un framework per l'apprendimento automatico (al momento mancante)
- Dati annotati da lettori esperti
- Software per visualizzare i risultati

Software Open Source

- Insight Toolkit (ITK) 
- Visualization Toolkit (VTK) 
- Cross Platform Make CMake 
- Fast Artificial Neural Network (FANN)
- SVMLIGHT
- Osirix 

Insight Toolkit ITK

- Framework per l'analisi delle immagini orientato alle immagini mediche
- Completamente scritto in C++
- Sistema di build CMake
- Progetto open source
- Documentazione + doxygen disponibili online
- Community molto ampia
- Possibilità di contribuire al framework



Sito ufficiale: <http://www.itk.org>

Template

- ITK è un software strutturato a “template”
- I template sono il mezzo con cui il C++ implementa la programmazione generica
- Esempio la classe `itk::image`, che implementa un’immagine, è un template sulla dimensione e sul tipo di pixel
- Esempio l’istruzione

```
itk::Image<int,3>
```

- definisce un’immagine 3-dimensionale in cui i valori dei pixel sono interi

Smart pointer

- Si comportano come normali puntatori
- Si occupano automaticamente di liberare l'area di memoria a cui puntano una volta che sono “inutilizzati”
- Implementano un memory management tramite reference counting
- Un oggetto non più referenziato viene distrutto una volta usciti dallo scope

Perché SmartPointer?

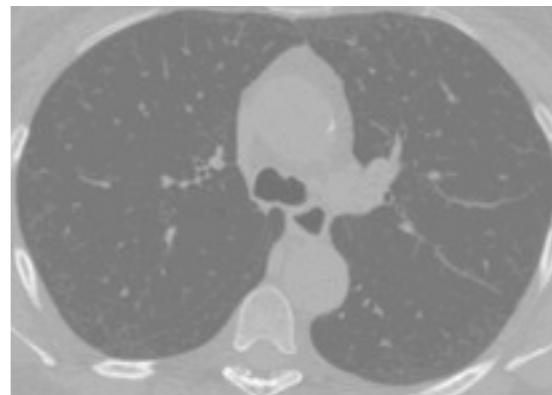
- Evitare i memory leak
- Evitare i double free
- Maggiore efficienza rispetto al garbage collecting in cui la memoria viene liberata a tempi non ben determinati

Perché è necessario lavorare con i puntatori?

- Le immagini digitali mediche sono tipicamente dell'ordine dei 200 Mb
- Passare una simile struttura dati per valore non è pensabile per ragioni performance e di consumo di memoria (copie degli oggetti e nuove allocazioni)

Esempio di ITK

Immagine di Input



Applico una
operazione di
thresholding

Filtro thresholding
binario di ITK



Immagine di
Output



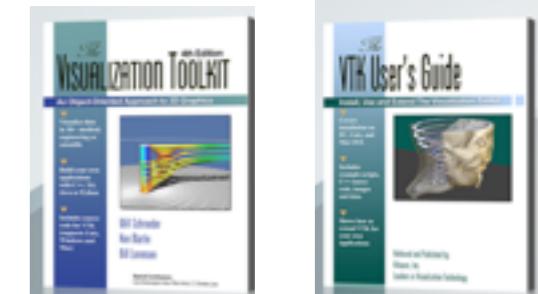
Esempio di codice

```
// Includo gli headers che contengono la classe itk Image
#include <itkImage.h>
#include <itkBinaryThresholdImageFilter.h>
#include <itkImageFileReader.h>
#include <itkImageFileWriter.h>
int main()
{
    // Assegno degli alias ai template che utilizzerò
    // immagine bidimensionale con pixel di tipo float
    typedef itk::Image<float,2> ImageType;
    // immagine bidimensionale con pixel di tipo int
    typedef itk::Image<short int,2> BinaryImageType;
    // oggetto che scrive una immagine di tipo BinaryImageType su disco
    typedef itk::ImageFileWriter< BinaryImageType > WriterType;
    // oggetto che legge una immagine di tipo ImageType da disco
    typedef itk::ImageFileReader< ImageType > ReaderType;
    // istanza della classe ImageType
    ReaderType::Pointer myReader=ReaderType::New();
    // leggo l'immagine da disco
    myReader->SetFileName("./input.gipl.gz");
    // Eseguo il filtro
    typedef itk::BinaryThresholdImageFilter< ImageType,BinaryImageType> BinaryThresholdFilterType;
    BinaryThresholdFilterType::Pointer myTsFilter=BinaryThresholdFilterType::New();
    myTsFilter->SetInput(myReader->GetOutput());
    myTsFilter->SetLowerThreshold(500);
    myTsFilter->SetUpperThreshold(10000);
    myTsFilter->SetInsideValue(0);
    myTsFilter->SetOutsideValue(1);
    // salvo il risultato su disco
    WriterType::Pointer myWriter=WriterType::New();
    myWriter->SetInput(myTsFilter->GetOutput());
    myWriter->SetFileName("./output.gipl.gz");
    myWriter->Update();
}
```



VTK

- Framework per la visualizzazione di immagini orientato alle immagini mediche
- Completamente scritto in C++ (binding anche per Python)
- Progetto open source
- Sistema di build CMake
- Documentazione tramite Doxygen e manuali (a pagamento)



Sito ufficiale: <http://www.vtk.org>

Esempio di cosa si può fare con VTK

Surface rendering di una maschera binaria



Volume rendering di un volume TAC





Cmake

- Sistema di build cross platform
- Supporta Linux/OSX e Windows
- E' una seria alternativa agli "autotools"
- Permette di scrivere progetti realmente cross platform astraendo dalle peculiarità del sistema tool di sviluppo usato

Sito ufficiale: <http://www.cmake.org/>

Come si scrive un progetto Cmake?

Contenuto del CMakeLists.txt

```
PROJECT(itkExample)
cmake_minimum_required(VERSION 2.6)
find_package(ITK)
# cerco ITK
IF(ITK_FOUND)
    include(${ITK_USE_FILE})
ELSE(ITK_FOUND)
    message(FATAL_ERROR "ITK not found. Please INSTALL IT or try to set ITK_DIR.")
ENDIF(ITK_FOUND)
# definisco la variabile ITK_LIBRARY che contiene tutte le librerie di ITK
set (ITK_LIBRARY ITKIO ITKCommon )
# definisco un eseguibile che verra' costruito con itkExample.cc
add_executable(itkExample itkExample.cc )
# definisco le librerie che linkeranno l'eseguibile
target_link_libraries(itkExample ${ITK_LIBRARY})
```

SVMLIGHT

- Progetto open source che implementa una SVM
- Scritto in C
- Supporta Solaris, Linux, MacOSX, Windows e Cygwin

Sito ufficiale: <http://svmlight.joachims.org/>

Fast Artificial Neural Network (FANN)

- Progetto open source che implementa una rete neurale multilayer
- Scritto in C
- Supporta Unix e Windows
- Forum e documentazione ampia

Sito ufficiale: <http://leenissen.dk/fann/wp/>



Osirix

- Software open source
- Scritto in objective-C
- Funziona soltanto sotto OSX
- Utilizza ITK e VTK
- Sito web e mailing list disponibili, ma documentazione scarsa (si impara soltanto copiando il codice altrui)

Sito ufficiale: <http://www.osirix-viewer.com/>

Parte III

Un esempio pratico, un sistema CAD per l'individuazione di noduli polmonari in immagini TAC

Acquisizione dei dati TAC

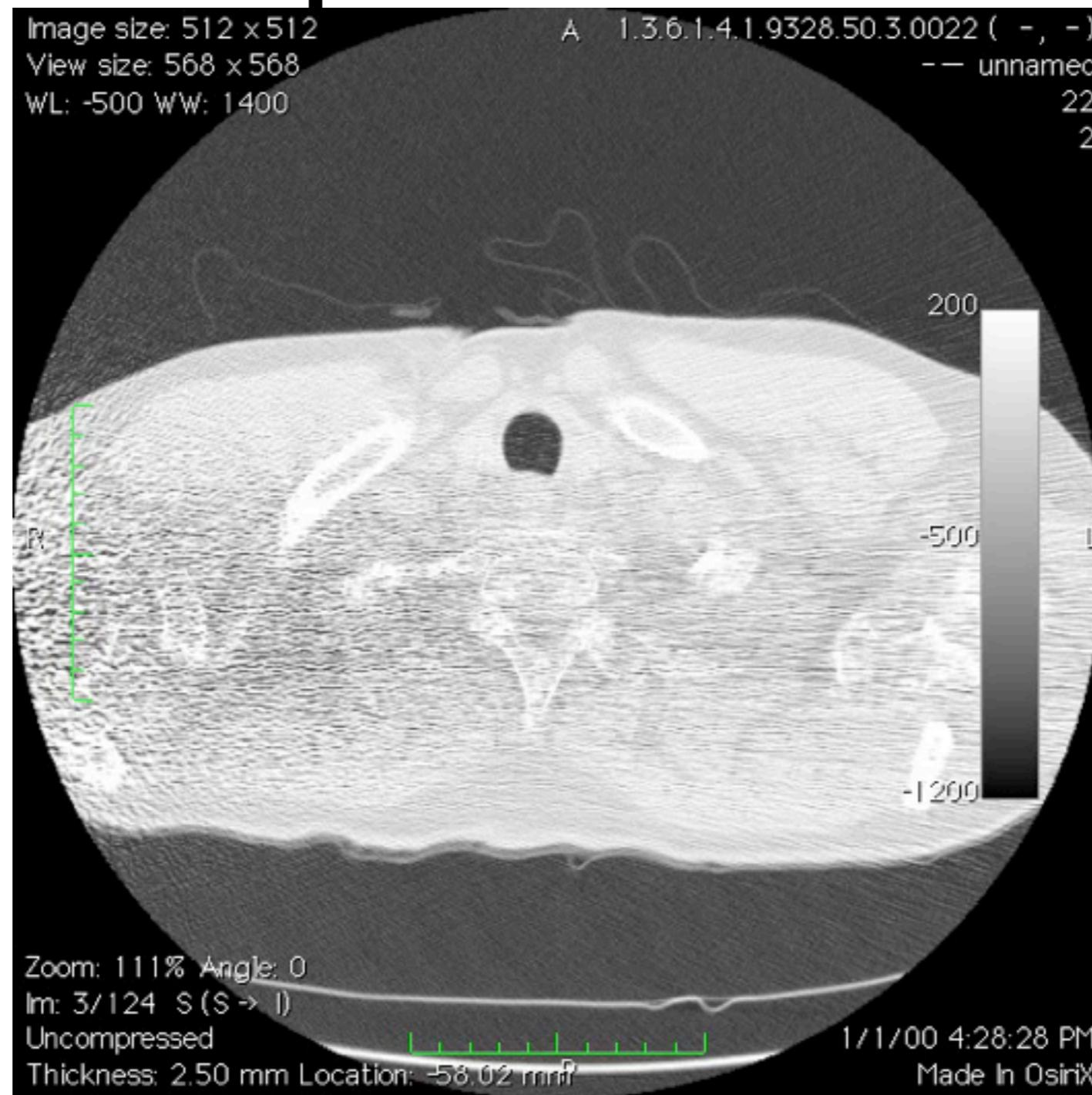
Attenuazione raggi-X attraverso il corpo umano



L'intero volume viene rappresentato da una serie di fette bidimensionali

Qual è l'aspetto di un esame polmonare CT?

Qual è l'aspetto di un esame polmonare CT?



Il problema del cancro ai polmoni

- Il cancro ai polmoni è tipicamente asintomatico nei primi stadi quando la cura può essere efficace
- Il tasso di sopravvivenza a 5 anni dalla diagnosi è solo del 10-15%
- Non ci sono stati negli ultimi 20 anni miglioramenti nell'aspettativa di vita di un paziente (al contrario di quello che avviene per gli altri cancri)
- La maggior parte dei cancri polmonari è diagnosticata troppo tardi perché il trattamento sia efficace
- I dati mostrano che cancri diagnosticati precocemente hanno un tasso di sopravvivenza che può raggiungere il 70%

[1] Micheli A, Baili P, Quinn M, Mugno E, Capocaccia R, Grosclaude P, EUROCARE Working Group. Life expectancy and cancer survival in the EUROCARE-3 cancer registry areas. Annals of oncology. 2003;14(Suppl 5):v28-v40.

[2]Ahmedin Jemal, DVM, PhD,Taylor Murray, Elizabeth Ward, PhD,Alicia Samuels, MPH, Ram C.Tiwari, PhD,Asma Ghafoor, MPH, Eric J. Feuer, PhD and Michael J.Thun, MD, MS,Cancer Statistics, 2005 CA cancer J Clin,55 10-30,2005

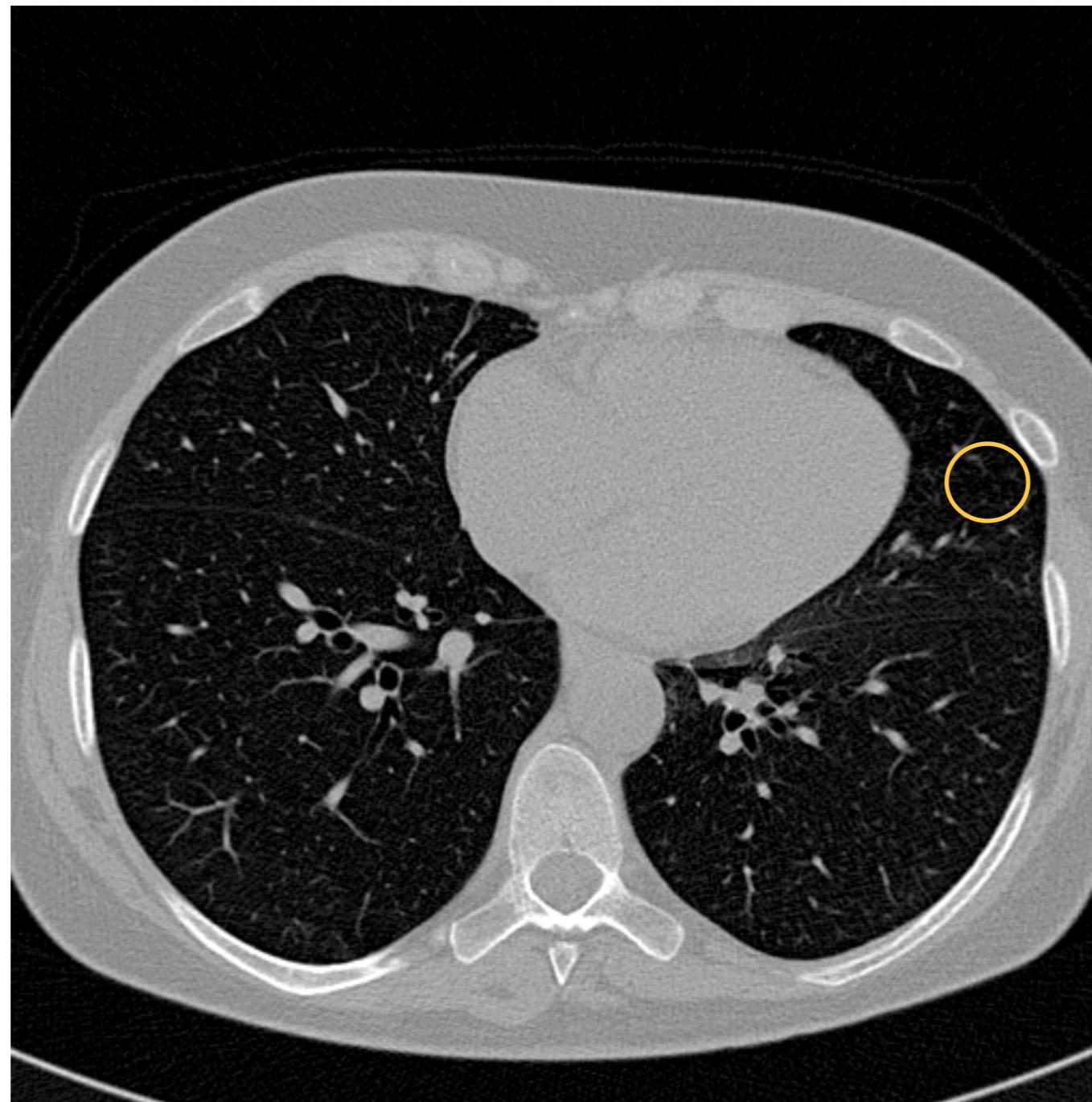
Screening trial del cancro ai polmoni con la CT

- Questi dati indicano che lo screening del cancro ai polmoni potrebbe essere utile nel ridurre la mortalità
- Su un campione di persone asintomatiche ad alto rischio si effettuano delle CT a bassa dose
- Ogni CT deve essere attentamente letta e annotata da un radiologo che deve annotare ogni nodulo, per eventualmente seguirne l'evoluzione temporale

Nodulo visto da una TAC

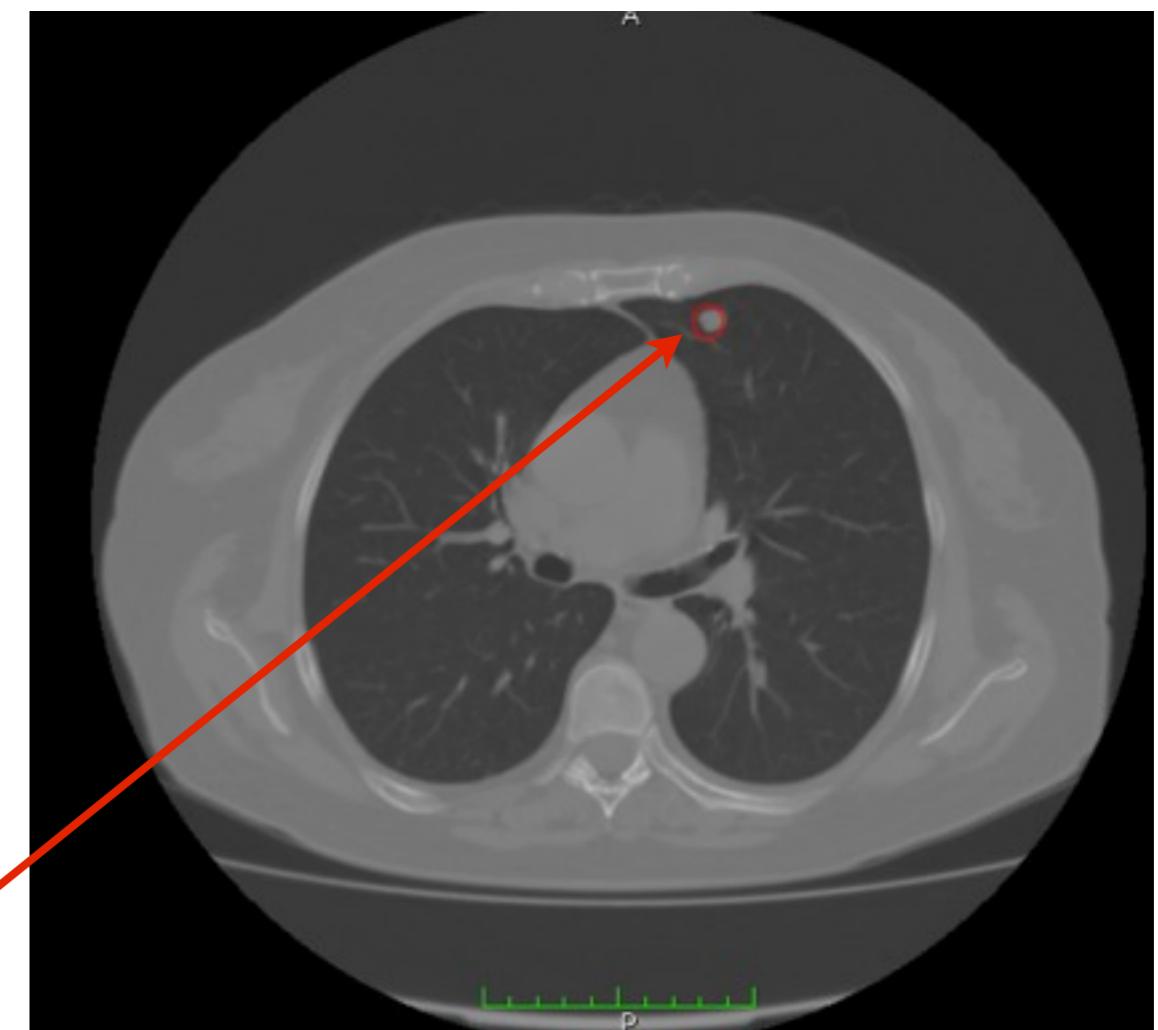


Nodulo visto da una TAC



Cosa stiamo cercando ?(1/2)

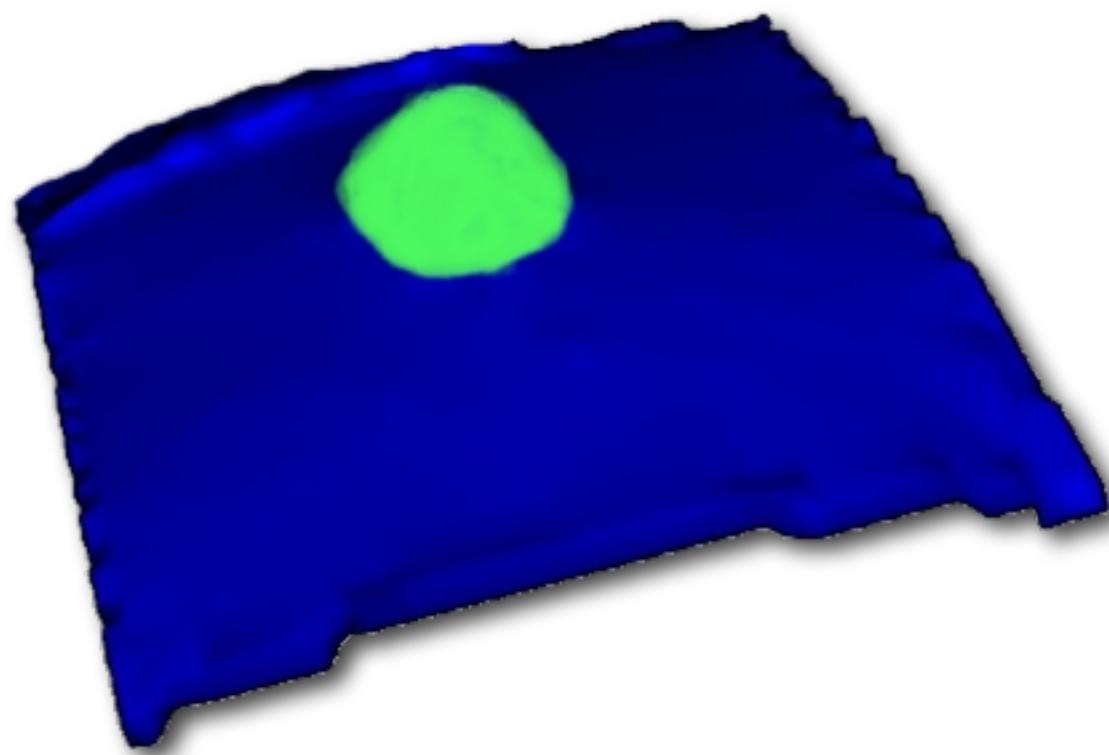
noduli interni



oggetti sferici completamente immersi nel parenchima polmonare

Cosa stiamo cercando? (2/2)

noduli pleurici



oggetti semisferici connessi alla superficie pleurica

The Voxel Based Neural Approach (VBNA) CADe

- Implementa due procedure complementari: una per i noduli interni (CAD_I) e l'altra per i noduli pleurici (CAD_{JP})
- I risultati dei due CAD sono poi uniti alla fine per ottenere i findings sull'intero polmone
- Interamente sviluppato con i software precedentemente elencati
- Gira correttamente sotto ubuntu linux e OSX

[1] "Multi-scale analysis of lung computer tomography images" I. Gori, F. Bagagli, M.E. Fantacci ,A. Preite Martinez,A. Retico, I. De Mitri, S. Donadio, C. Fulcheri, G. Gargano, R. Magro, M. Santoro, S. Stumbo Journal of Instrumentation, September 2007 - Volume 09, Issue 09, pp. P09007.

[2] Retico,A., Bagagli, F., Camarlinghi, N., Carpentieri, C., Fantacci, M. E., Gori, I., "A voxel-based neural approach (VBNA) to identify lung nodules in the ANODE09 study" SPIE Medical Imaging 2009

[3] N. Camarlinghi, I. Gori, A. Retico, F. Bagagli (2010). Selective reduction of CAD false-positive findings. In: Proceedings of SPIE Medical Imaging 2010, vol. 7624, p. 762430, doi: 10.1117/12.844432

CAD_I

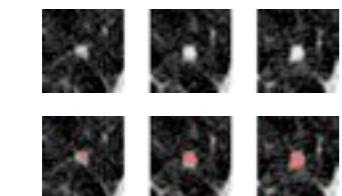
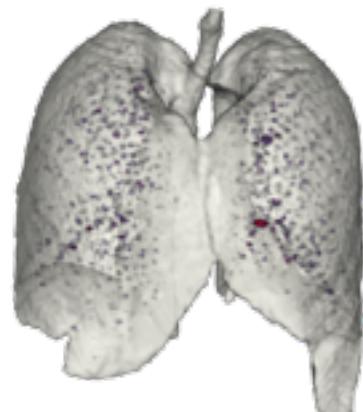
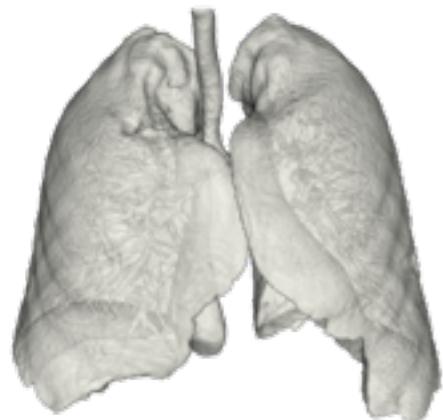
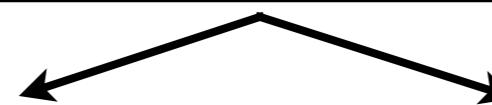


Immagine TAC



Segmentazione del polmone



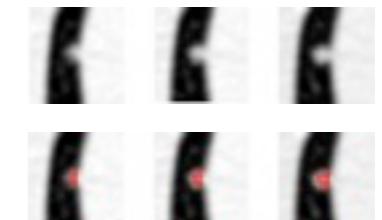
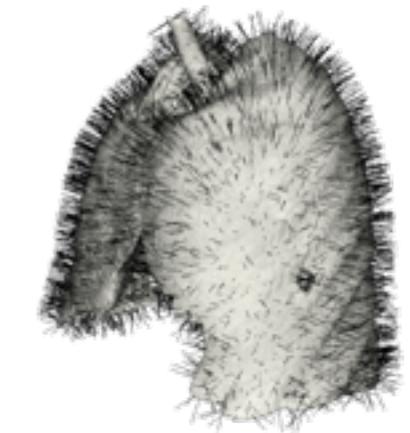
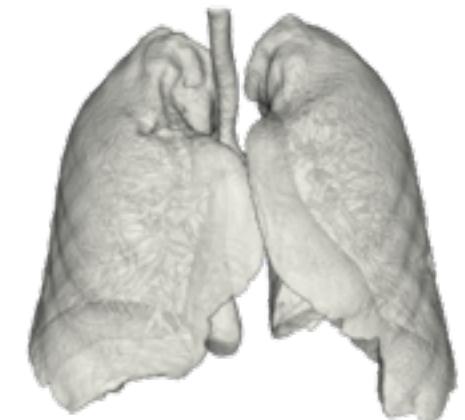
Cercatore di candidati noduli interni

Cercatore di candidati noduli pleurici

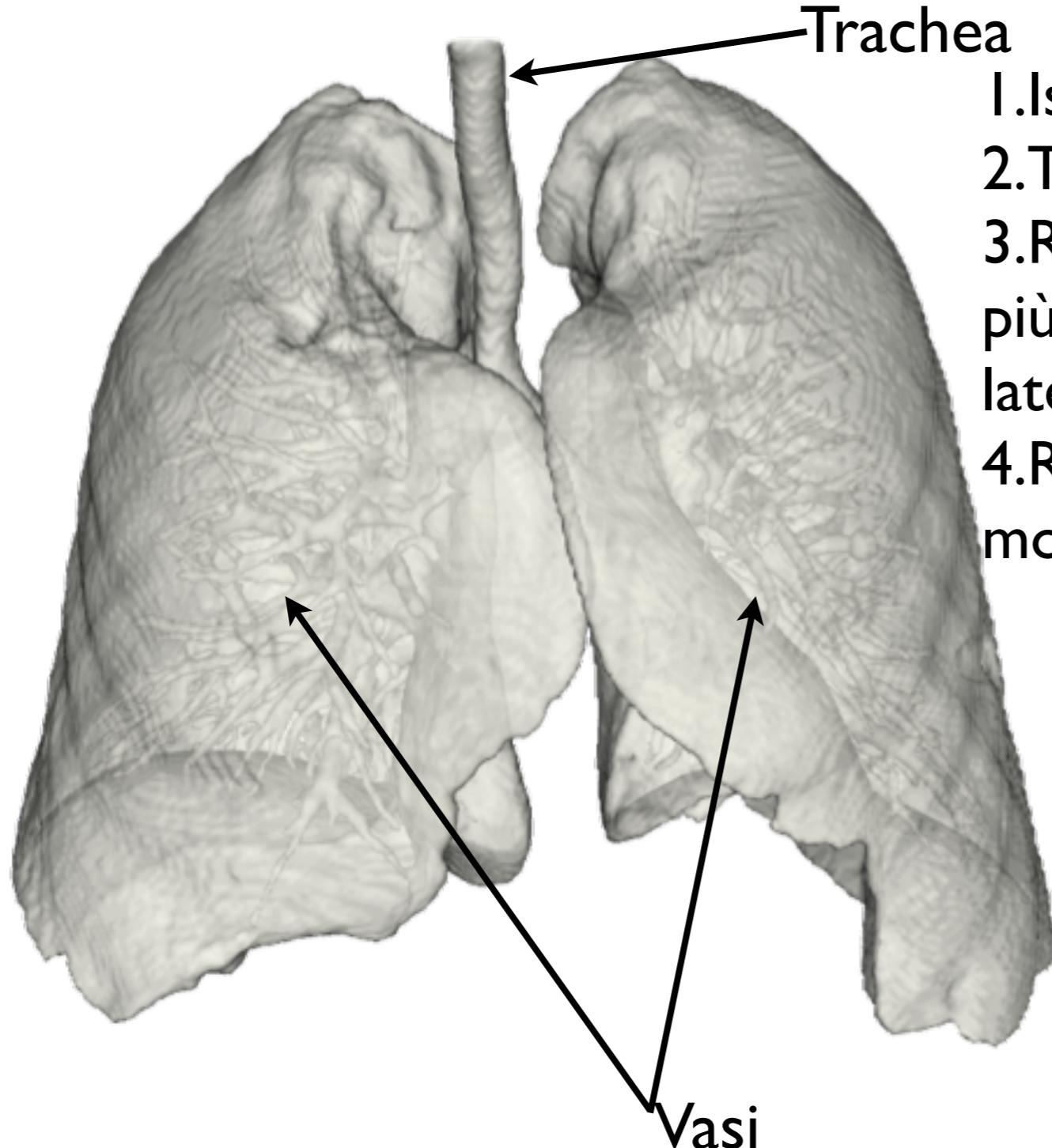
Riduzione Falsi positivi VBNA

Riduzione Falsi positivi VBNA

CAD_{JP}



Segmentazione polmonare



1. Isotropizzazione dell'immagine
2. Thresholding a [-1000,-600] HU
3. Ricerca della componente connessa più grande che non intersechi i bordi laterali dell'immagine
4. Rimozione dei vasi con operatori morfologici

Identificazione di candidati noduli interni

1.Si effettua uno smoothing gaussiano
dell'immagine a diverse scale

$$\sigma_i = \sigma_0, \dots, \sigma_1$$

2.Si valutano gli autovalori della matrice
Hessiana $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3|$

3.Si calcola il valore

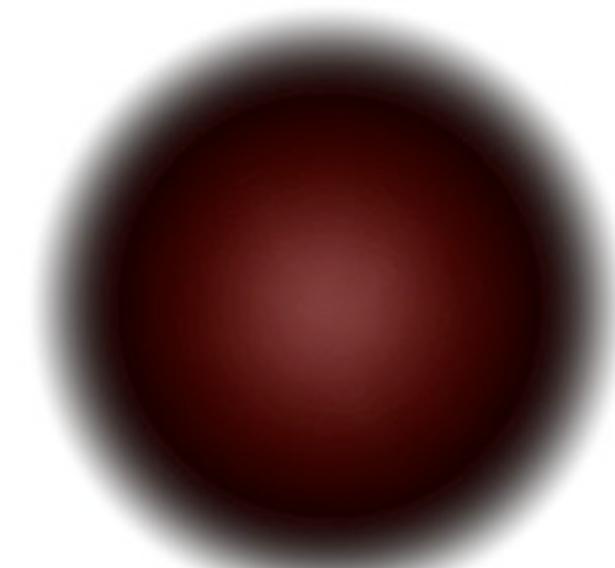
$$z_{dot} = \begin{cases} \frac{|\lambda_3|^2}{|\lambda_1|} & if \lambda_1, \lambda_2, \lambda_3 < 0 \\ 0 & otherwise \end{cases}$$

4.Si calcola il valore

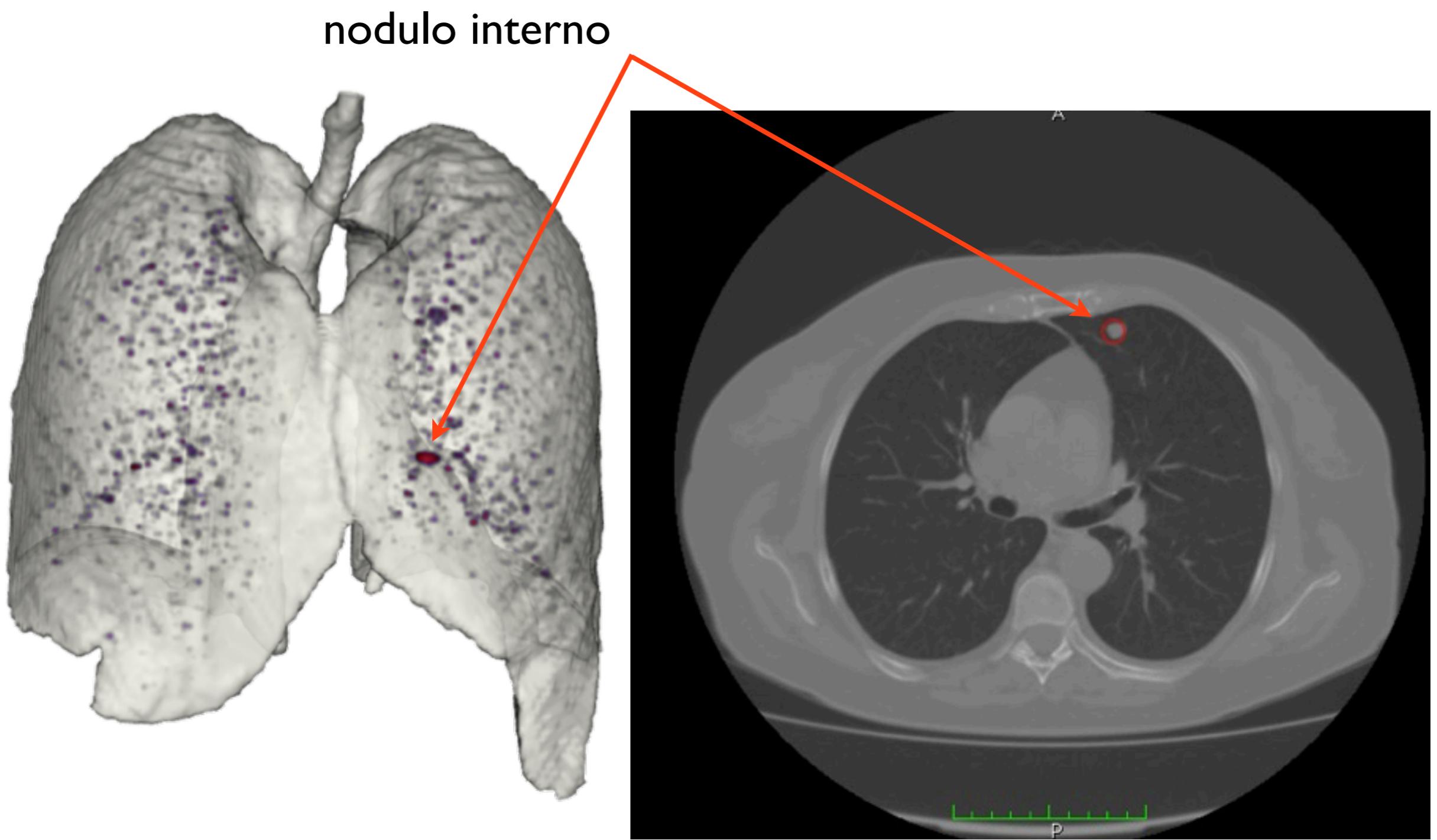
$$z_{max} = \max_i(\sigma_i^2 z_{dot}(\sigma_i))$$

5.I candidati noduli interni sono i massimi
locali nella matrice z_{max}

**I noduli interni sono
modellizzati come
distribuzioni gaussiane
3-dimensionali di
intensità di grigio**

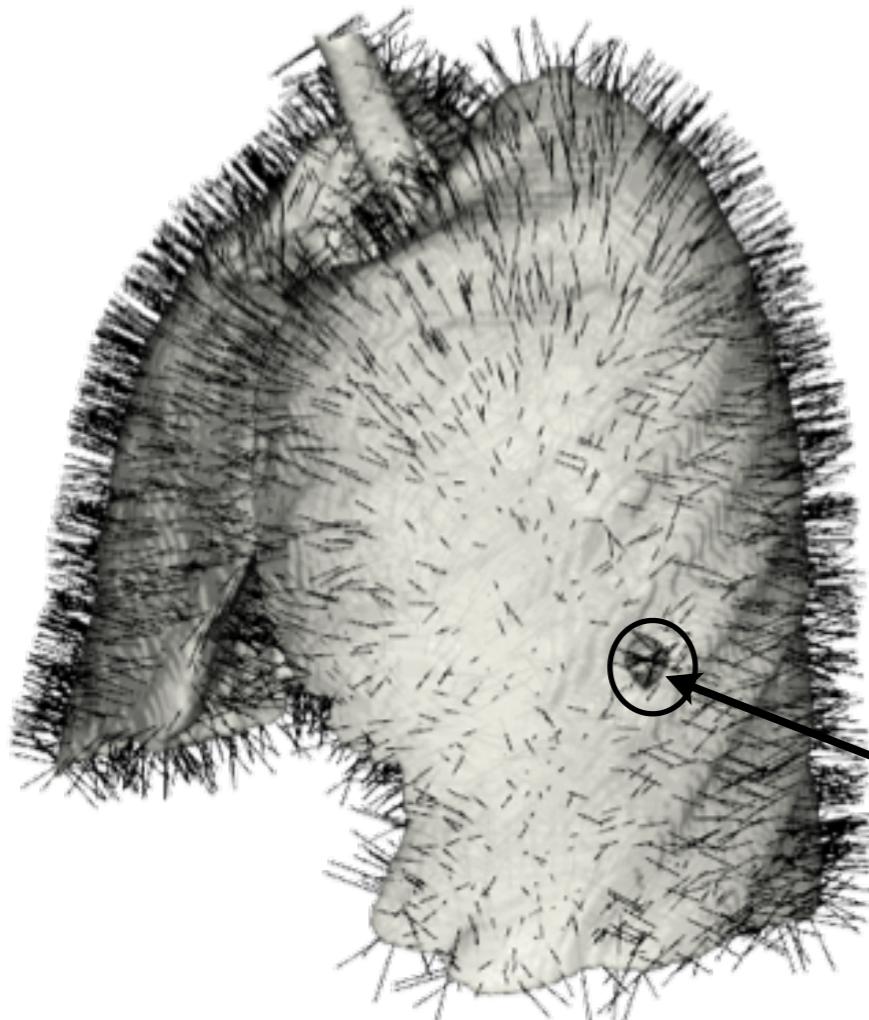


Esempio di output

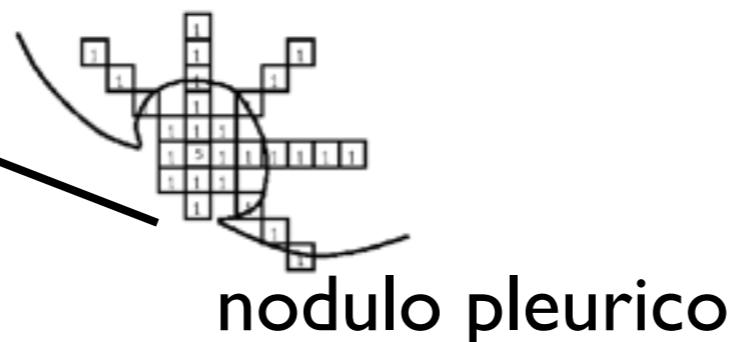


Identificazione di candidati noduli pleurici

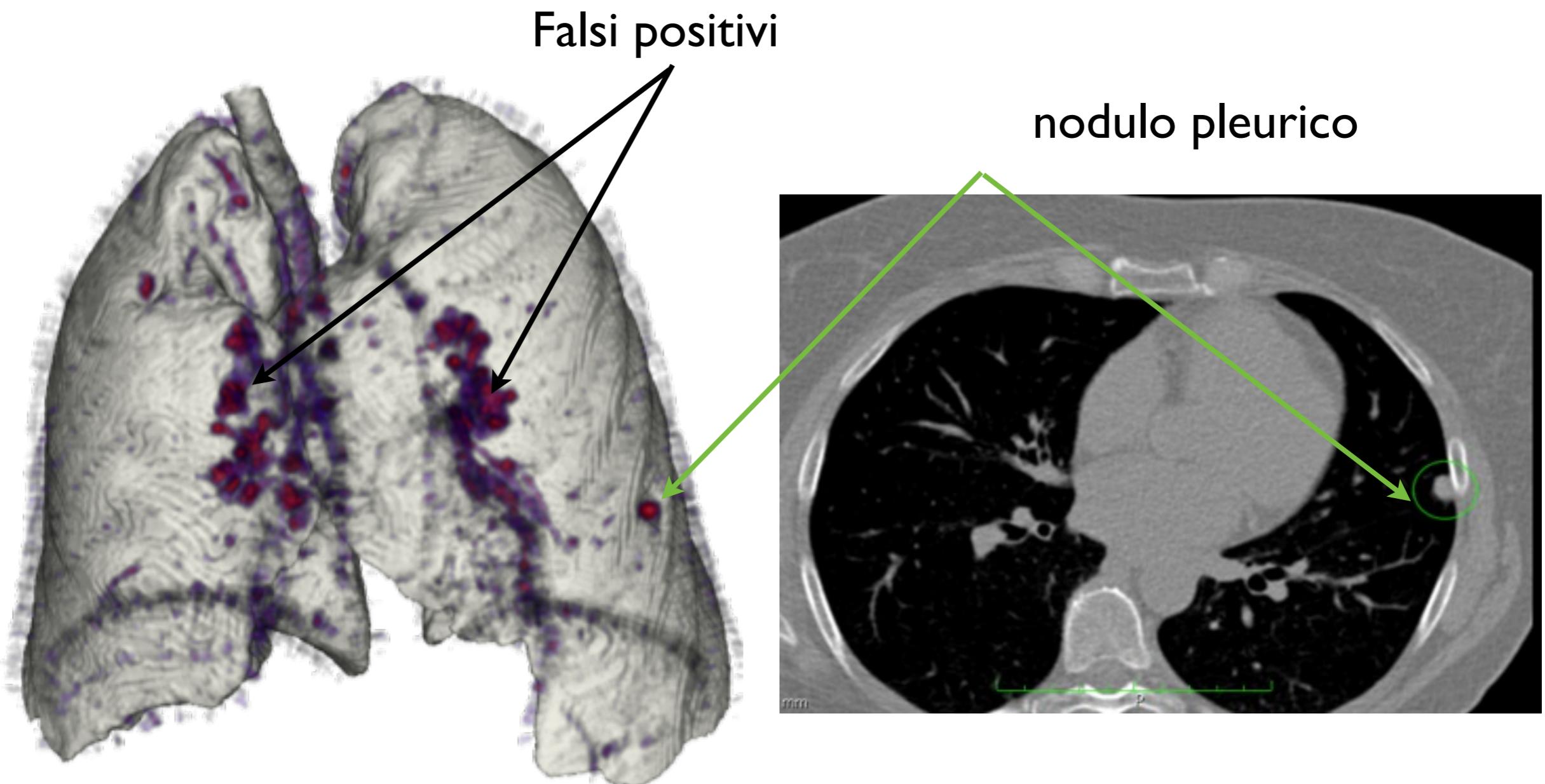
Normali alla superficie pleurica



1. Stiamo cercando oggetti semisferici connessi alla superficie pleurica
2. I centri di questi oggetti saranno punti in cui “molte normali” si intersecano
3. Cercando i massimi locali nella matrice che conta quanti segmenti normali si intersecano in ogni voxel abbiamo i candidati noduli



Esempio di output

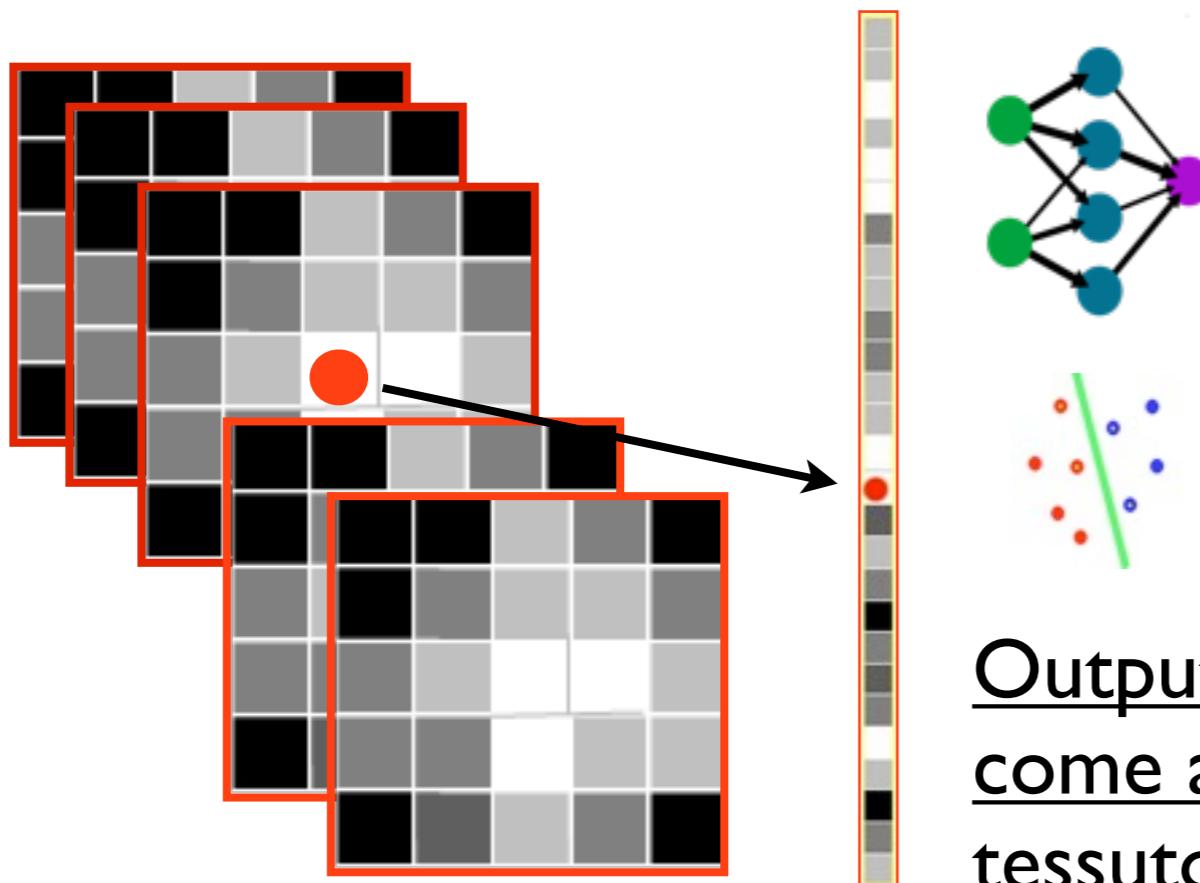


I falsi positivi sono tipicamente imperfezioni della pleura e vasi non rimossi efficientemente dalle procedure

Voxel Based Neural Approach (VBNA) riduzione dei falsi positivi

Partendo dalla lista di candidati del CAD_I e del CAD_{JP}, estraiamo tutti i voxel.

Per ciascun voxel appartenente a ciascun candidato estraiamo il seguente vettore di features

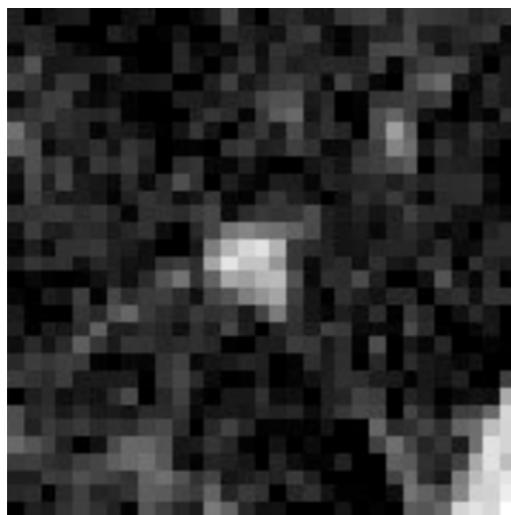


5x5x5 valori di intensità dell'intorno + gli autovalori dell'Hessiano + gli autovalori della matrice gradiente (tot 131 features)

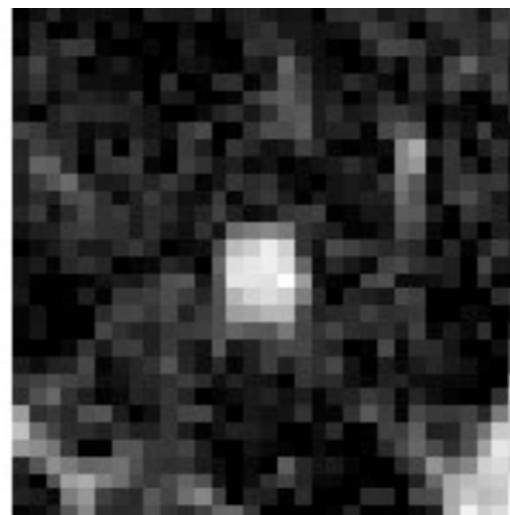
Output: un voxel può essere classificato come appartenente a un nodulo o a tessuto normale

Esempio di classificazione: nодulo interno (1/3)

$z-1$



z



$z+1$

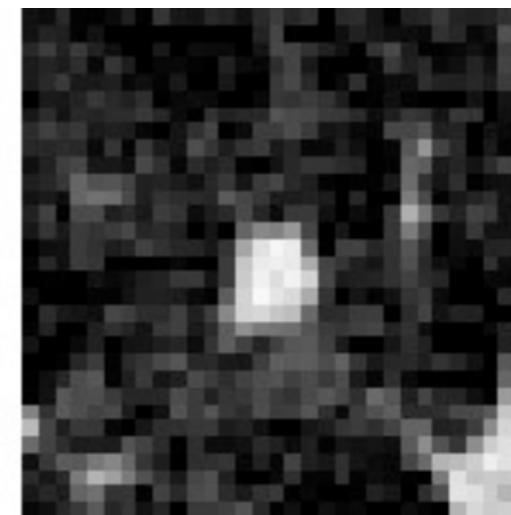
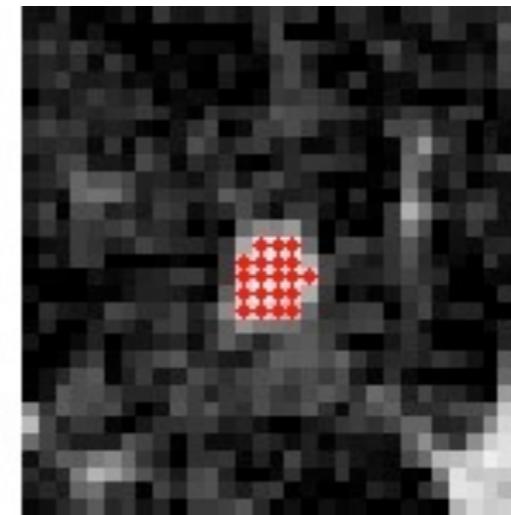
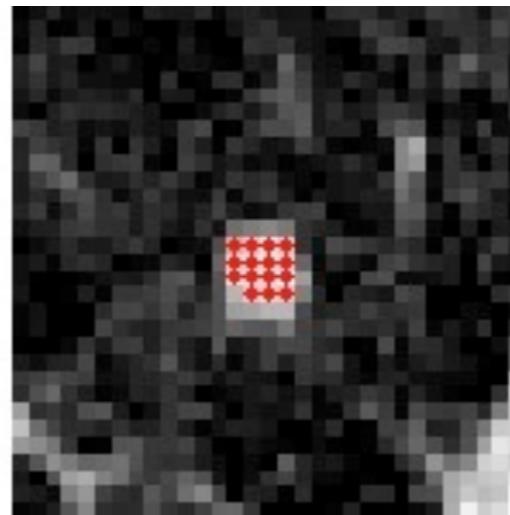
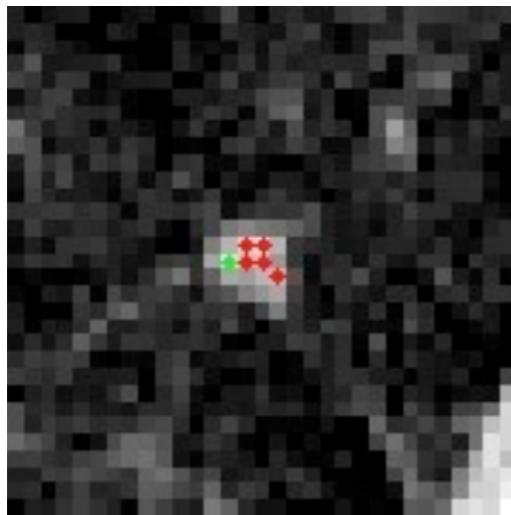


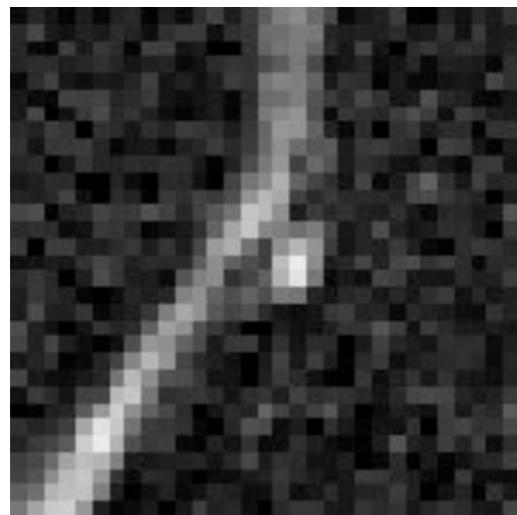
Immagine
originale



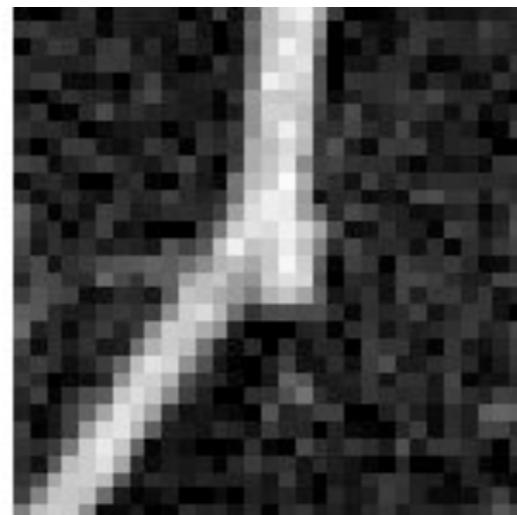
Normal Tissue	
Nodule Tissue	

Esempio di classificazione falso positivo(2/3)

$z-1$



z



$z+1$

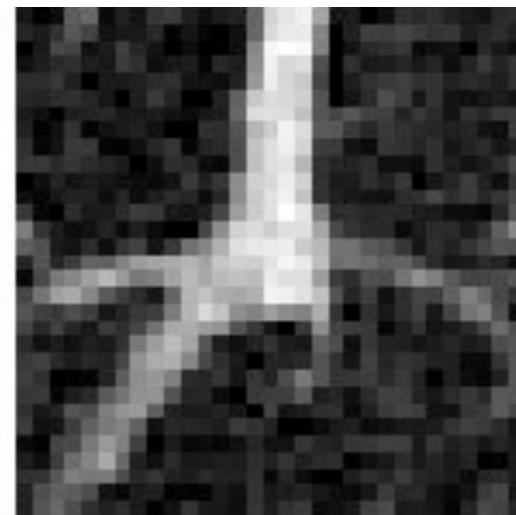
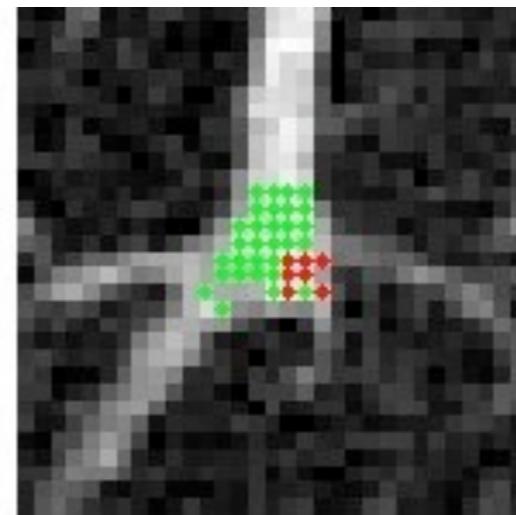
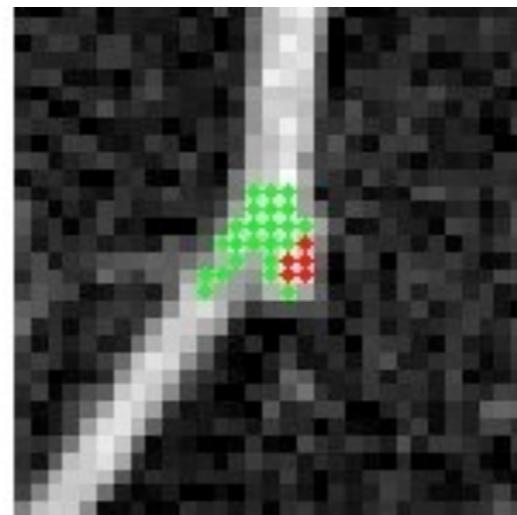
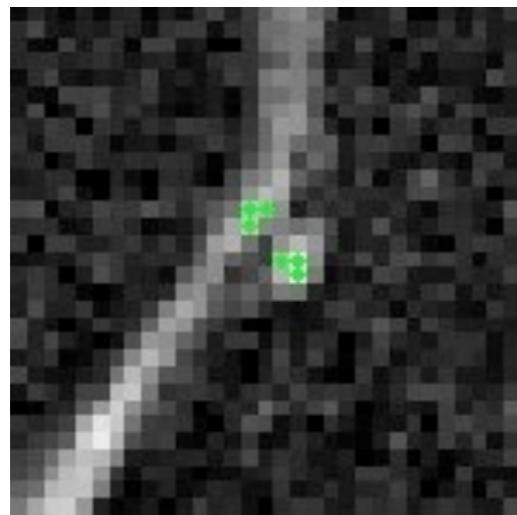


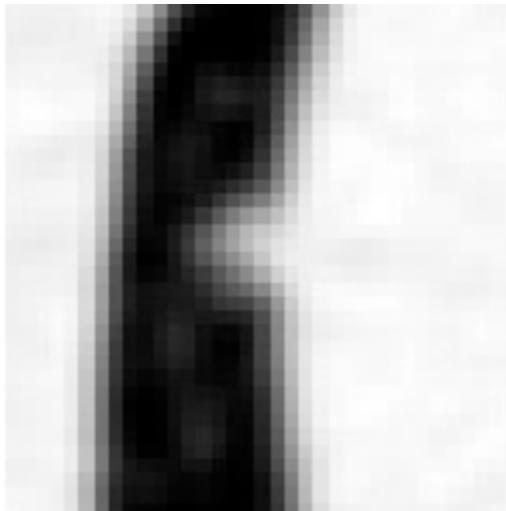
Immagine
originale



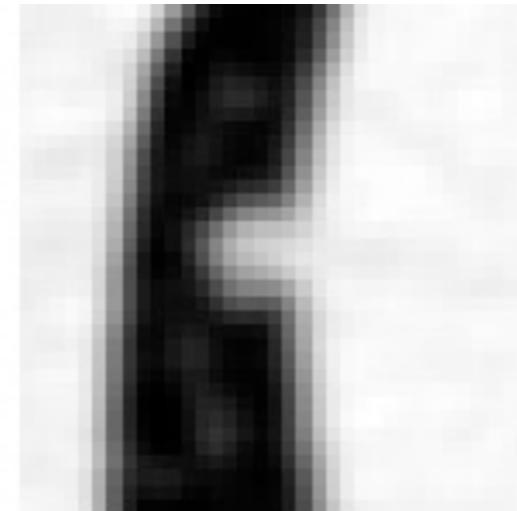
Normal Tissue	
Nodule Tissue	

Esempio di classificazione noduli pleurico (3/3)

$z-1$



z



$z+1$

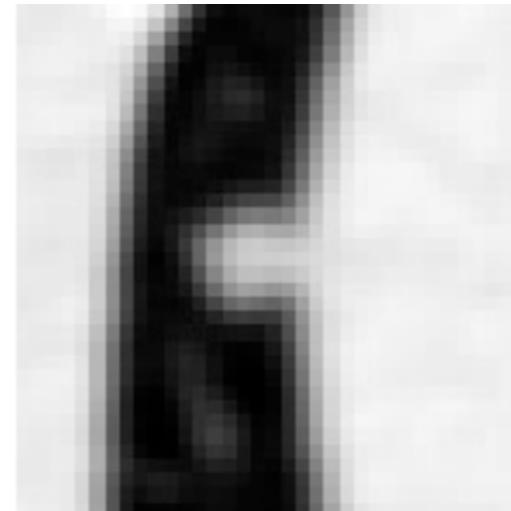
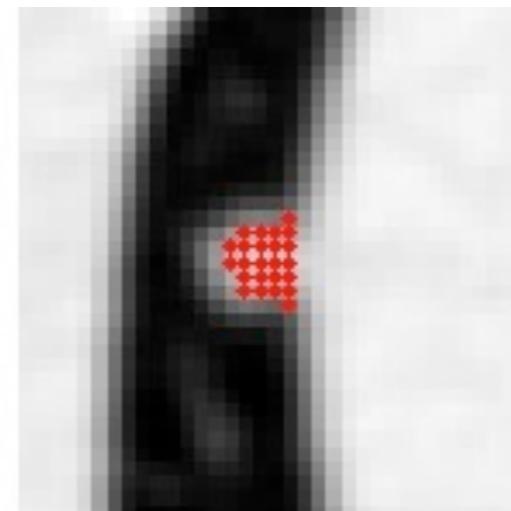
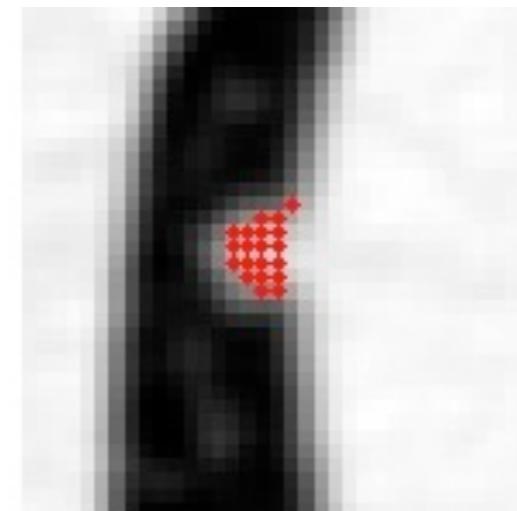
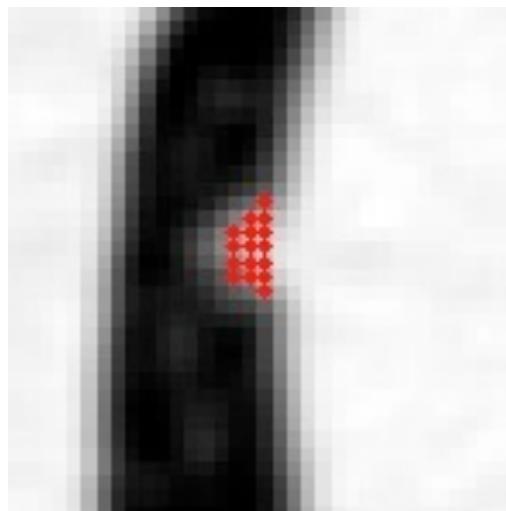


Immagine
originale



Normal Tissue	Green
Nodule Tissue	Red

Conclusioni

- Abbiamo affrontato varie tematiche nel campo delle immagini mediche
- Abbiamo inoltre visto un esempio di applicazione sviluppata completamente con software opensource
- Spero di avervi dato un'idea del ruolo che riveste (o dovrebbe rivestire) l'open source nella comunità scientifica

Grazie per l'attenzione!!!

Ringraziamenti

Si ringraziano: il Dr. F. Falaschi, e il Dr. C. Spinelli dell’“Unità Operativa Radiodiagnostica 2” dell’ “Azienda Ospedaliera Universitaria Pisana” e il Prof. D. Caramella e M. Barattini della Unità di “Radiologia Diagnostica e Interventistica” dell’Università di Pisa che hanno reso disponibile il database delle scansioni CT. Inoltre si ringraziano tutti i membri della collaborazione MAGIC-V.