Image Segmentation

Christian Salvatore
Scuola Universitaria Superiore IUSS Pavia

christian.salvatore@iusspavia.it

Image Segmentation

Definition

Image segmentation is a computer vision process that involves dividing an image into distinct, meaningful, and semantically homogeneous regions or objects.

This process aims to partition an image into non-overlapping segments, where each segment corresponds to a particular object, region, or feature within the image.

The goal is to simplify the representation of an image, making it easier to analyze and extract relevant information.

Segmentation : partitioning of an image I into different subregions S

classes

$$I = \sum_{i} S_i \qquad I = 1,\dots n$$

$$S_i \subset I \qquad S_i \cap S_j = \emptyset$$

Image Segmentation

Object classification is a subtask of object recognition, focusing specifically on assigning a label or category to a detected object based on its visual features. It involves categorizing objects into predefined classes or groups.

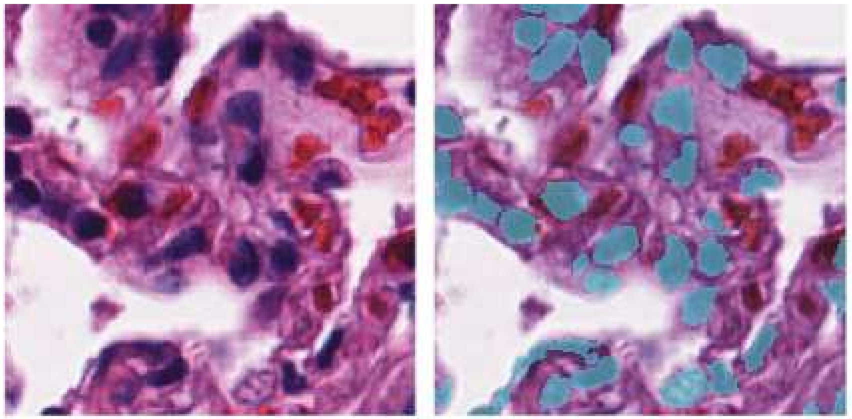
Object recognition is the task of identifying, categorizing objects within an image or a scene. This goes beyond detection by assigning a semantic label to each detected object, specifying its class or category.

Object localization refers to the process of determining the spatial location of an object within an image. This involves specifying the coordinates (usually represented by a bounding box) that encloses the object, indicating its position and extent.

Object detection is a computer vision task that involves identifying and localizing multiple objects within an image or a video. The goal is to detect the presence of objects, typically belonging to predefined classes, and provide bounding boxes around each detected object.

Image Segmentation

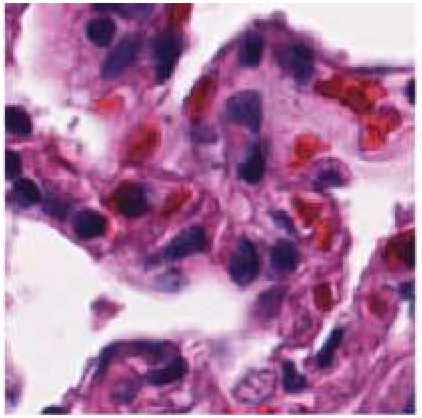
Semantic segmentation involves classifying each pixel in an image into predefined categories or classes, without distinguishing between individual instances of objects. The goal is to assign a semantic label to every pixel, indicating the type of object or region it belongs to. This technique provides a holistic understanding of the image content by segmenting it into semantically meaningful parts.



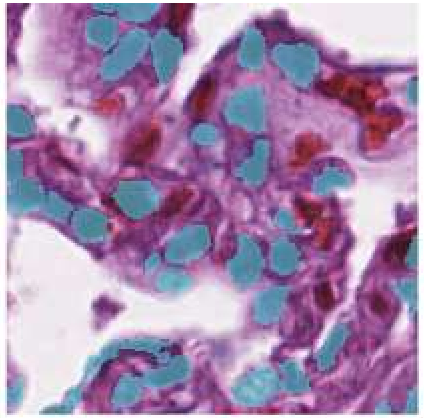
(a) Image (b) Semantic Segmentation

Image Segmentation

Semantic segmentation involves classifying each pixel in an image into predefined categories or classes, without distinguishing between individual instances of objects. The goal is to assign a semantic label to every pixel, indicating the type of object or region it belongs to. This technique provides a holistic understanding of the image content by segmenting it into semantically meaningful parts.

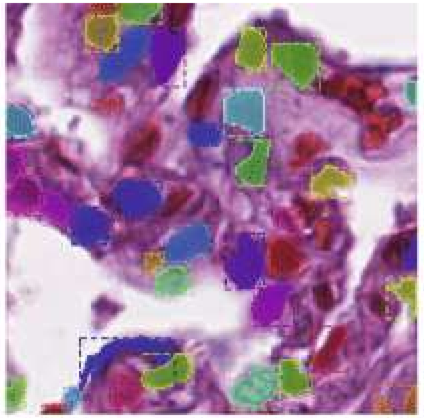


(a) Image



(b) Semantic Segmentation

Instance segmentation goes a step further than semantic segmentation by not only classifying pixels into categories but also distinguishing between individual instances of objects within the same class. In other words, it identifies and delineates each distinct object instance separately. This level of segmentation is valuable in scenarios where precise object boundaries and counting instances are crucial.



(c) Instance Segmentation

Image Segmentation

Semantic segmentation involves classifying each pixel in an image into predefined categories or classes, without distinguishing between individual instances of objects. The goal is to assign a semantic label to every pixel, indicating the type of object or region it belongs

to. the me

Panoptic segmentation is a computer vision task that entails jointly addressing semantic segmentation and instance segmentation. It involves partitioning an image into coherent regions, where each pixel is labeled either with a semantic category for amorphous elements (stuff) or with a unique instance identifier for countable and distinct objects (things). This unified segmentation framework aims to provide a comprehensive understanding of visual scenes by combining both semantic and instance-level information, facilitating detailed analysis and interpretation of complex visual content.

Ins sel int inc

other words, it identifies and delineates each distinct object instance separately. This level of segmentation is valuable in scenarios where precise object boundaries and counting instances are crucial.

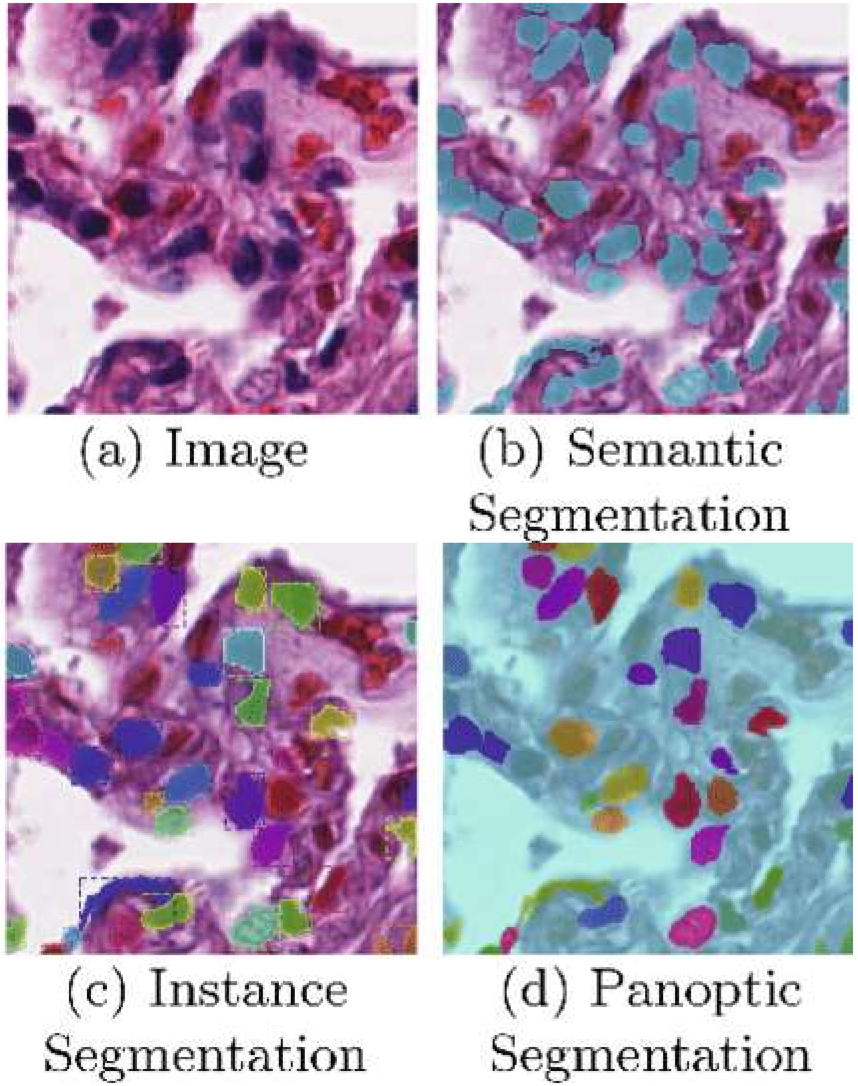
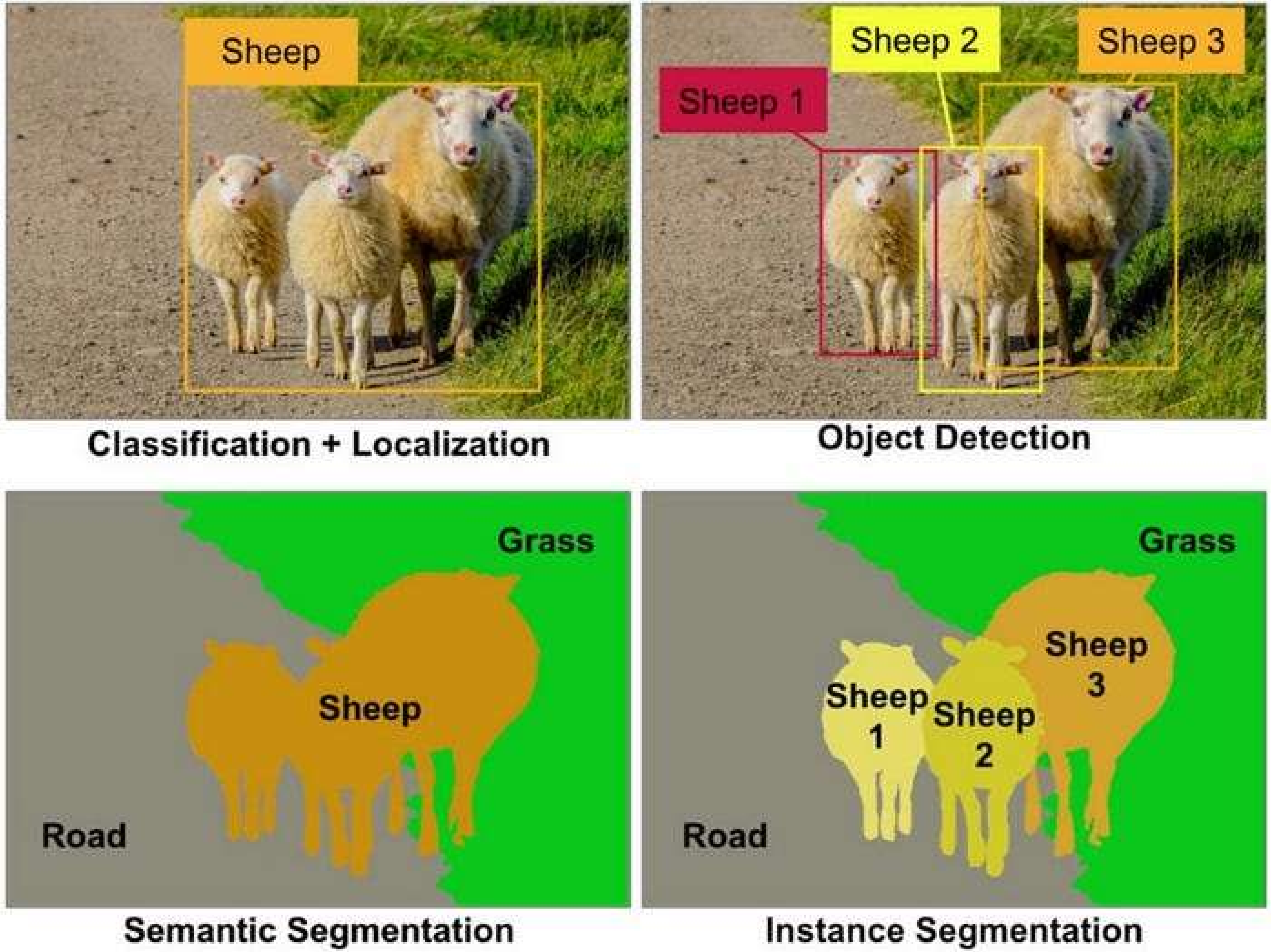
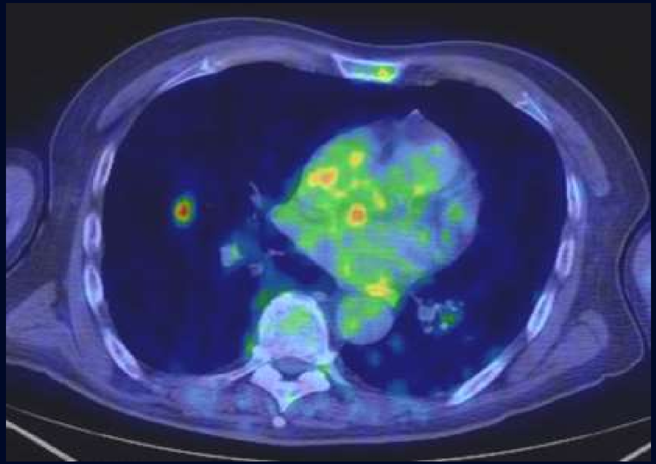


Image Segmentation



Medical-image Segmentation



Tumour

Complex biological object

Macroscopically

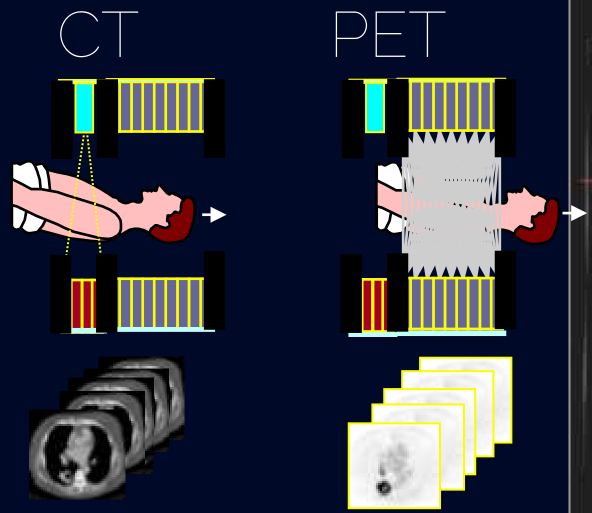
Heterogeneaus shape, density, global metabolism

Microscopically

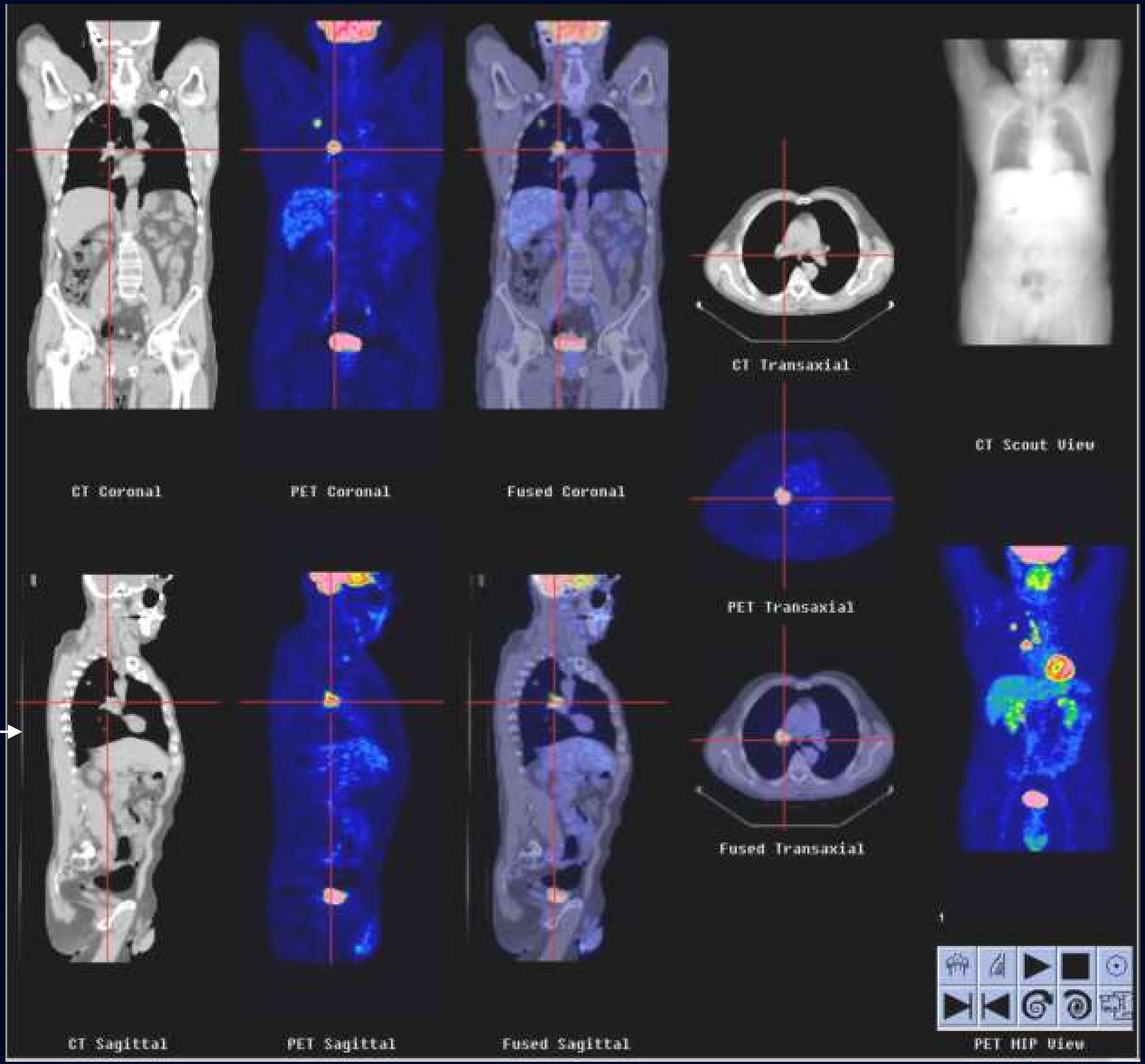
Cell proliferation, hipoxia, noeangiogenesis

PET/CT in oncology

[18F] FDG PET-CT



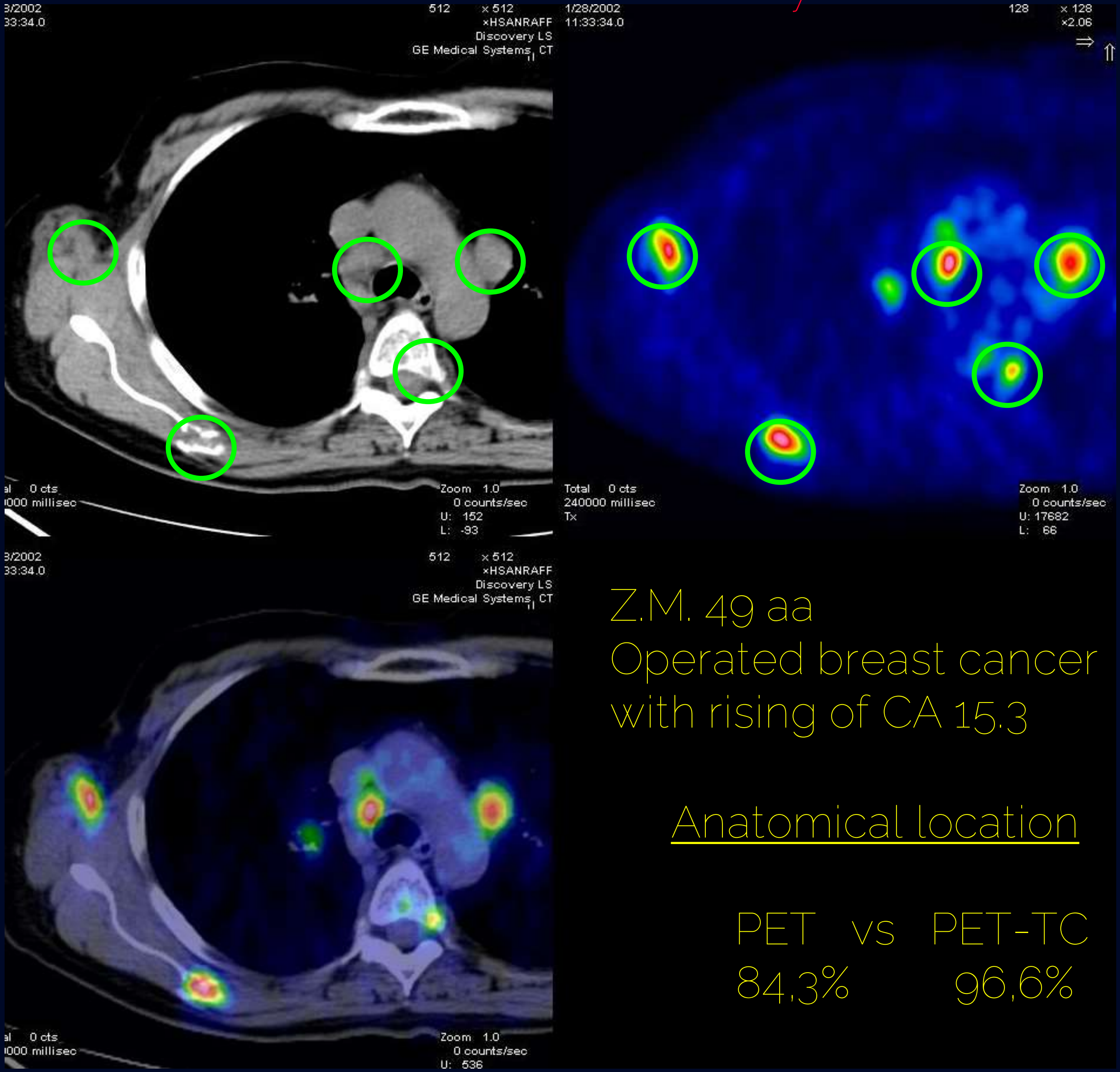
HSR Milan



PET/CT in oncology

- ANATOMICAL LOCATION
- CHARACTERIZATION
- CHOICE OF THERAPY
- THERAPY MONITORING

Qualitative analysis



Z.M. 49 aa
Operated breast cancer with rising of CA 15.3

Anatomical location

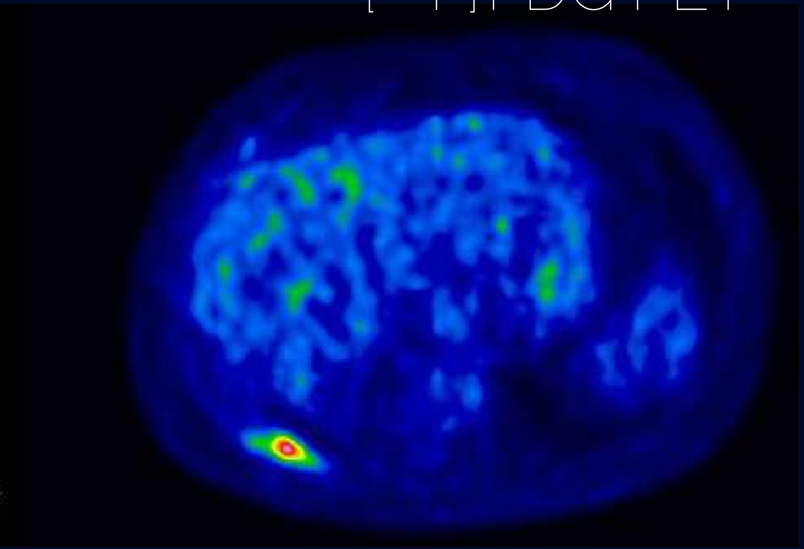
PET	vs	PET-TC
84,3%		96,6%

HSR Milan

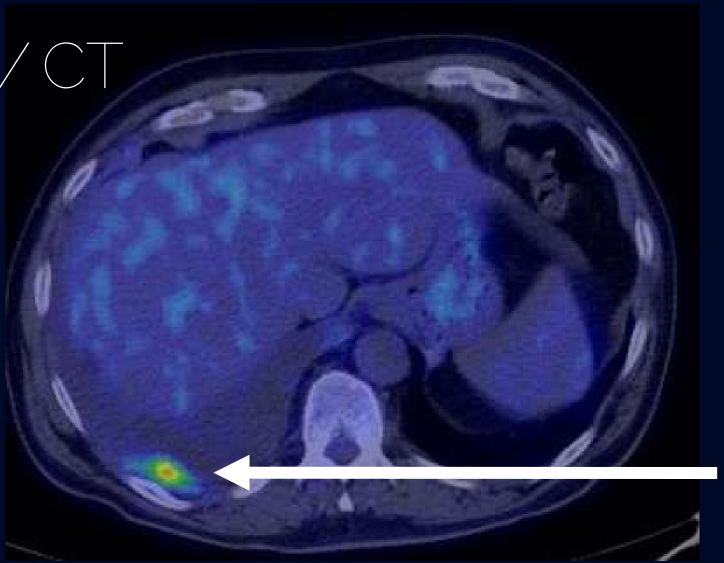
Qualitative analysis

CT

[^{18}F]FDG PET



PET/ CT



PET-CT Characterization:
Lung cancer

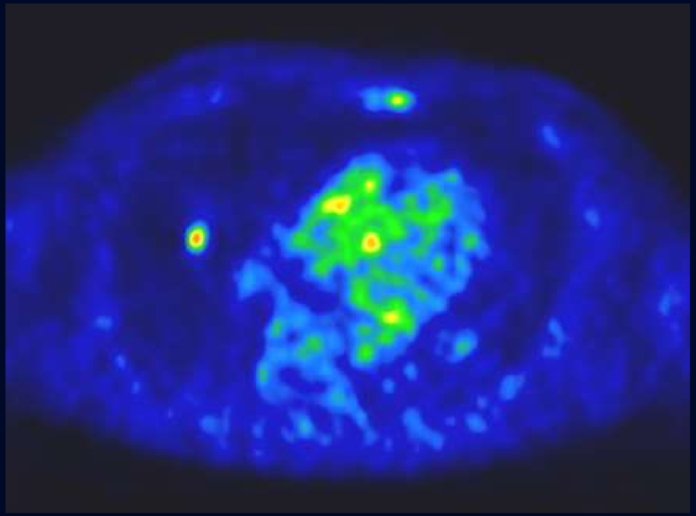
M (Pleural mts)

HSR Milan

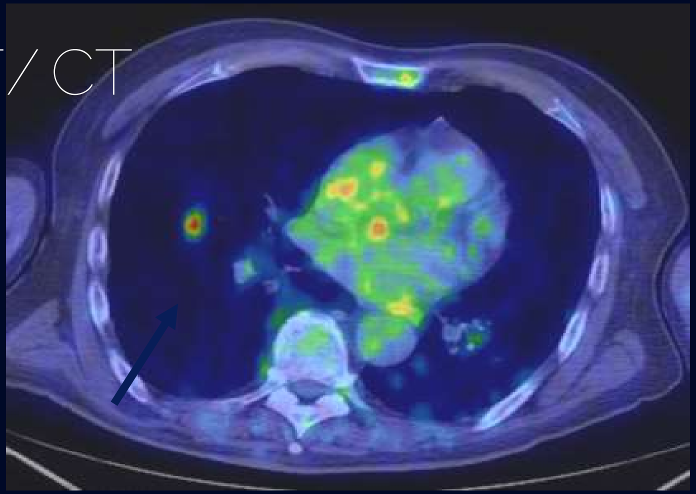
Qualitative analysis

CT

[18F] FDG PET



PET / CT



MF-50 yrs.
Exclusion from radiotherapy treatment for presence of lung metastasis

HSR Milan

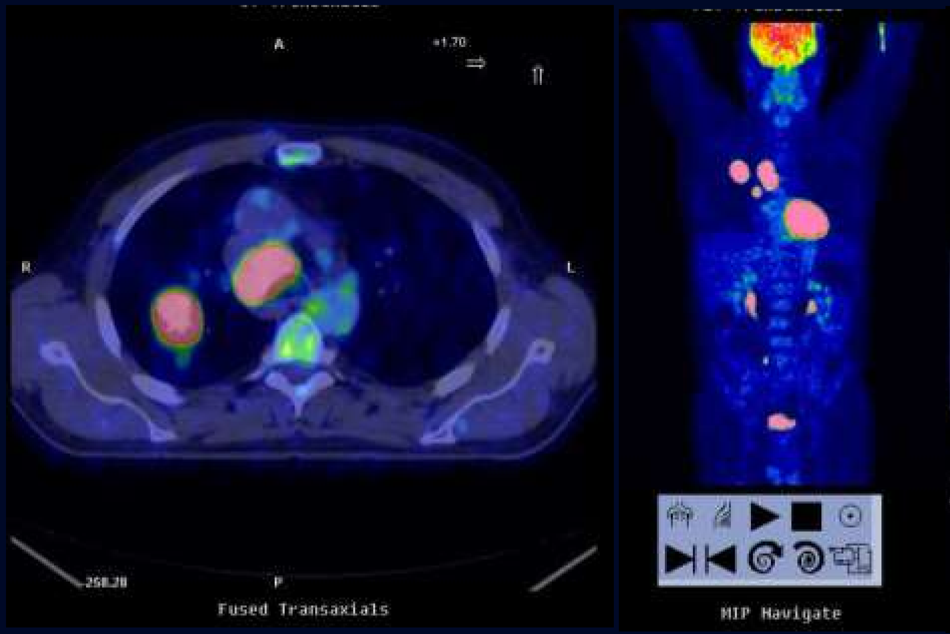
Qualitative analysis

$[^{18}F]$FDG PET/ CT

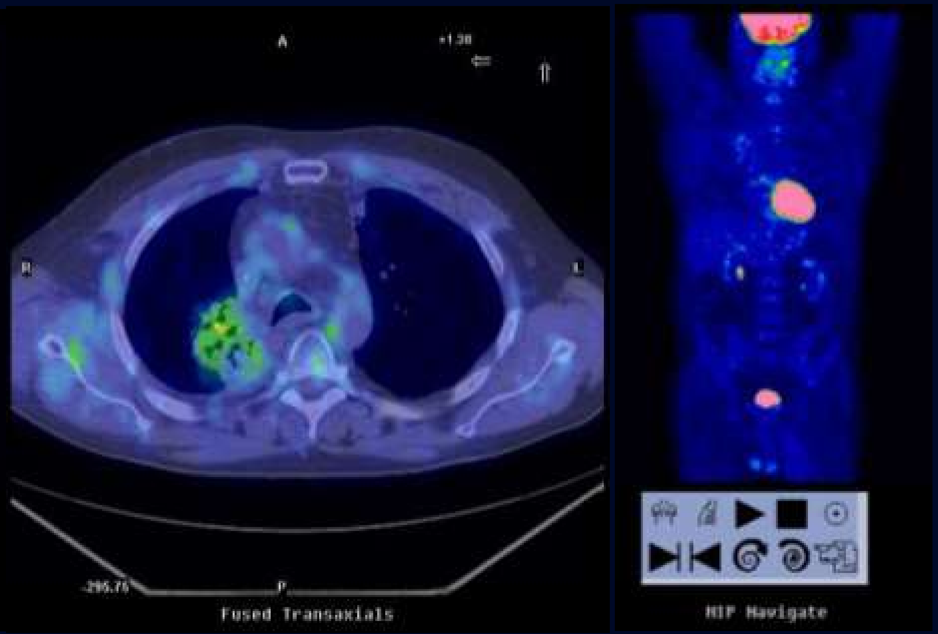
Evaluation of residual mass

Evaluation of recurrence

Evaluation of therapy (early and at the end of treatment)



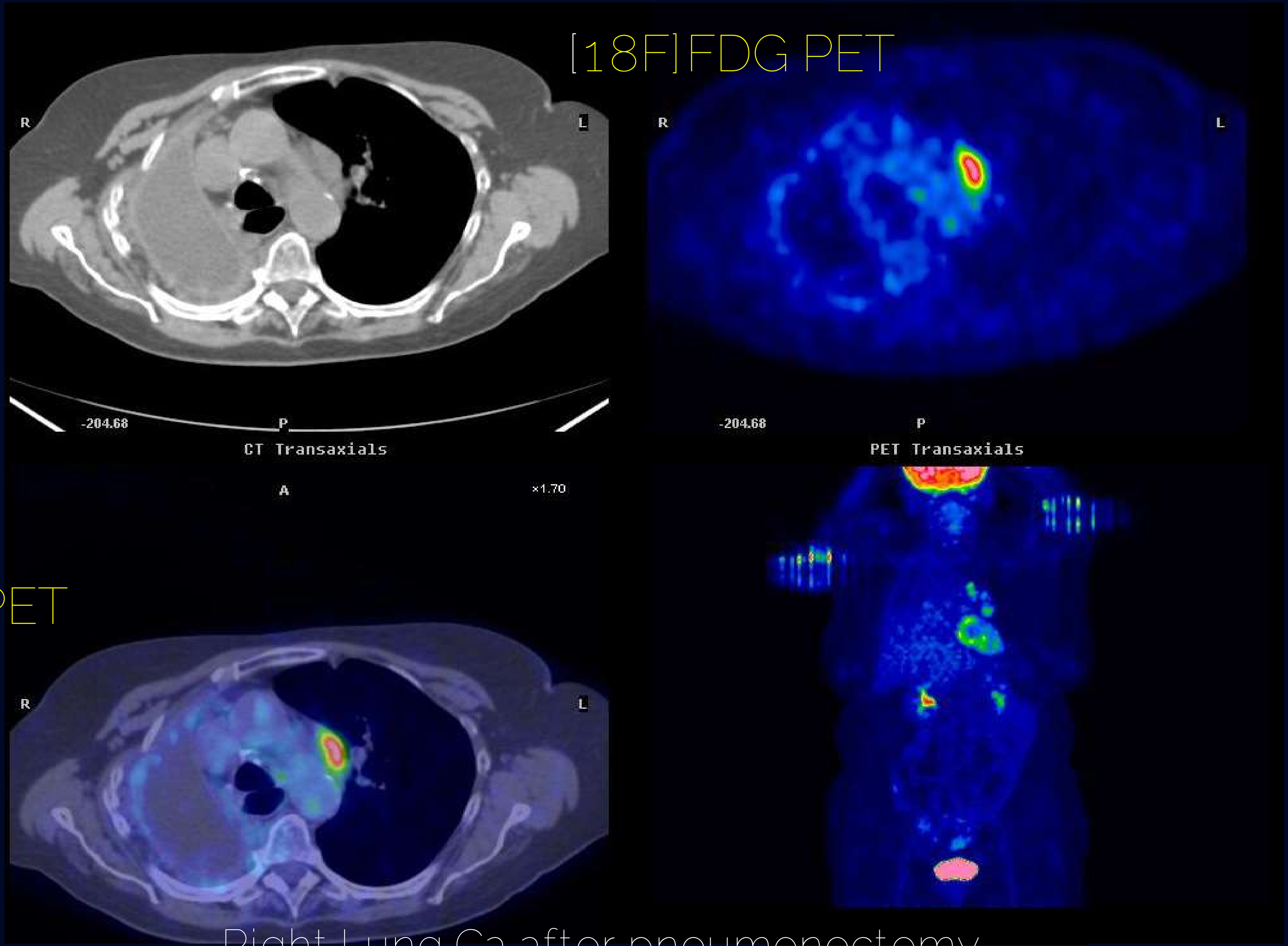
Before Therapy



After Therapy

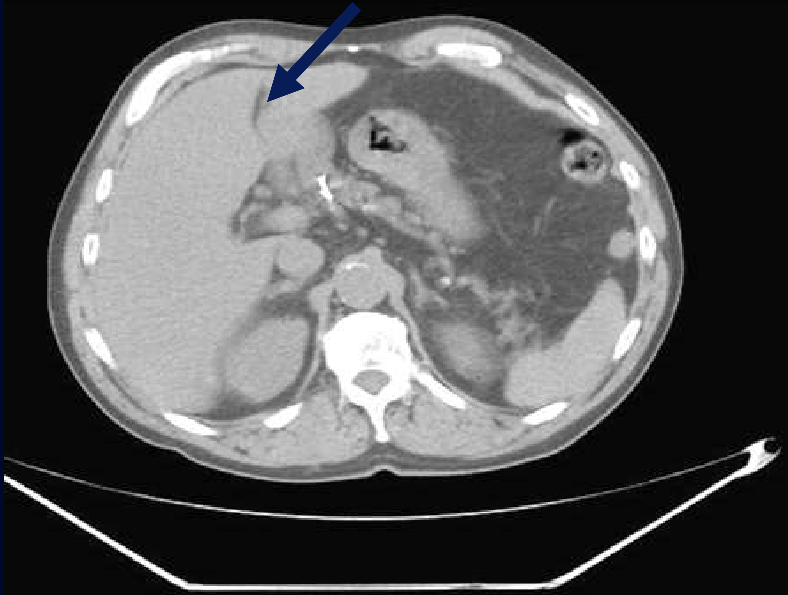
Qualitative analysis

Residual mass

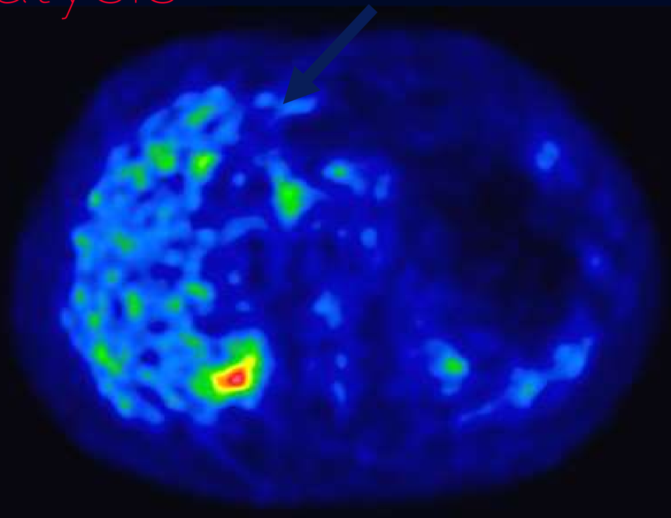


Right Lung Ca after pneumonectomy

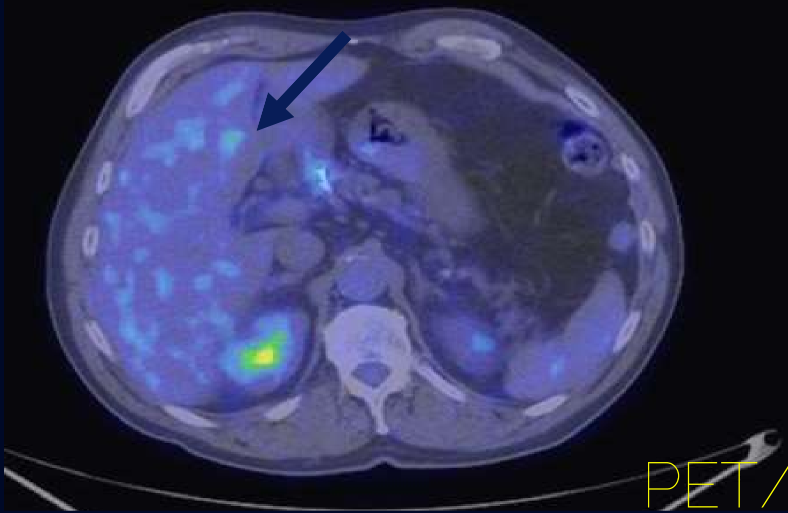
Qualitative analysis



CT



[18F]FDG-PET



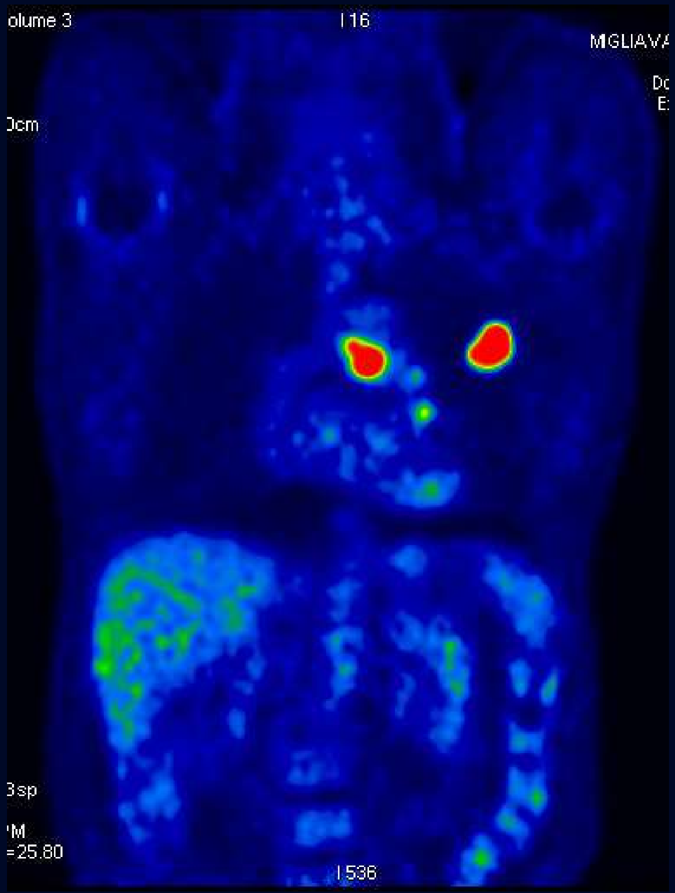
PET/CT

E.B. 56 aa
22/4/2002

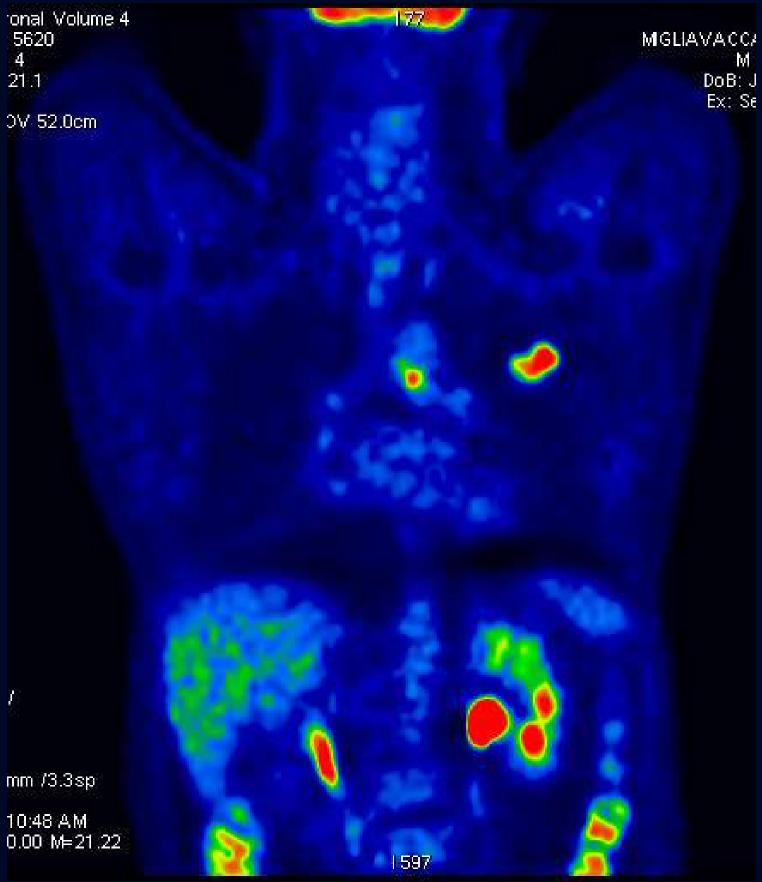
Pancreatic Ca after surgery

Recurrence

Qualitative analysis
Monitoring therapy response



Pre Chemotherapy Study
Apr 03 2007



Post Chemotherapy Study
Sept 27 2007

ROI SEGMENTATION

Qualitative methods → Visually inspection,

Manual contouring

Quantitative methods → Image processing,

semi-auto/autocontouring

Manual methods

- Image modality-dependence
- Operator-dependence
- Window level-dependence (colorbar)

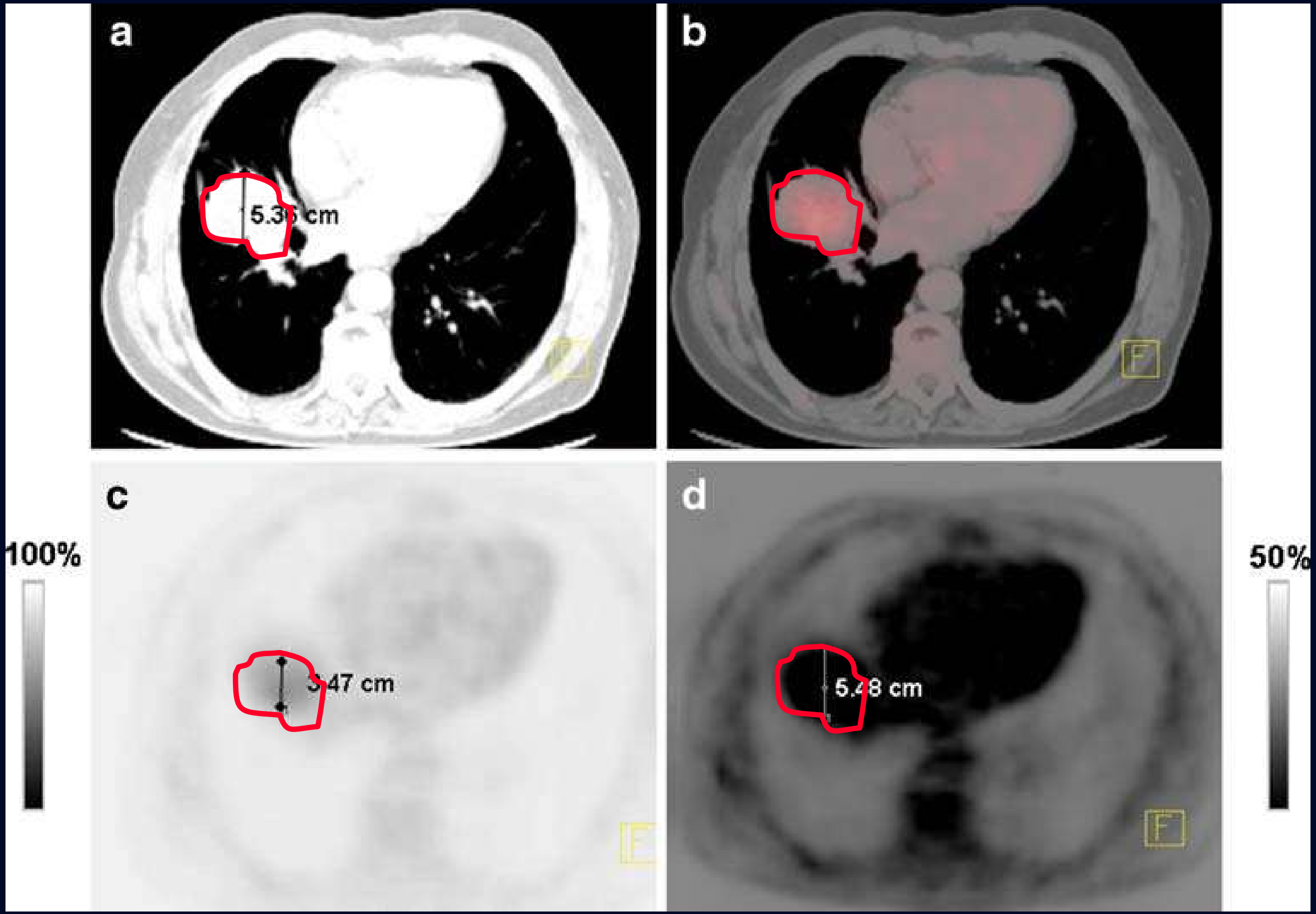
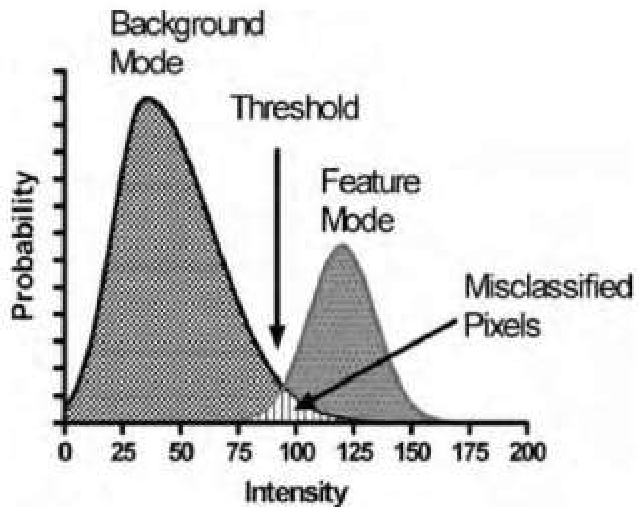
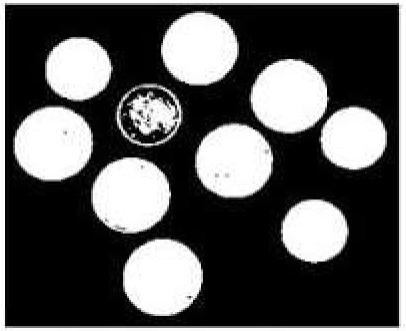


Image Segmentation

Thresholding is a straightforward image segmentation method that involves setting a threshold value. Pixels with intensity values above or below this threshold are allocated to distinct regions.

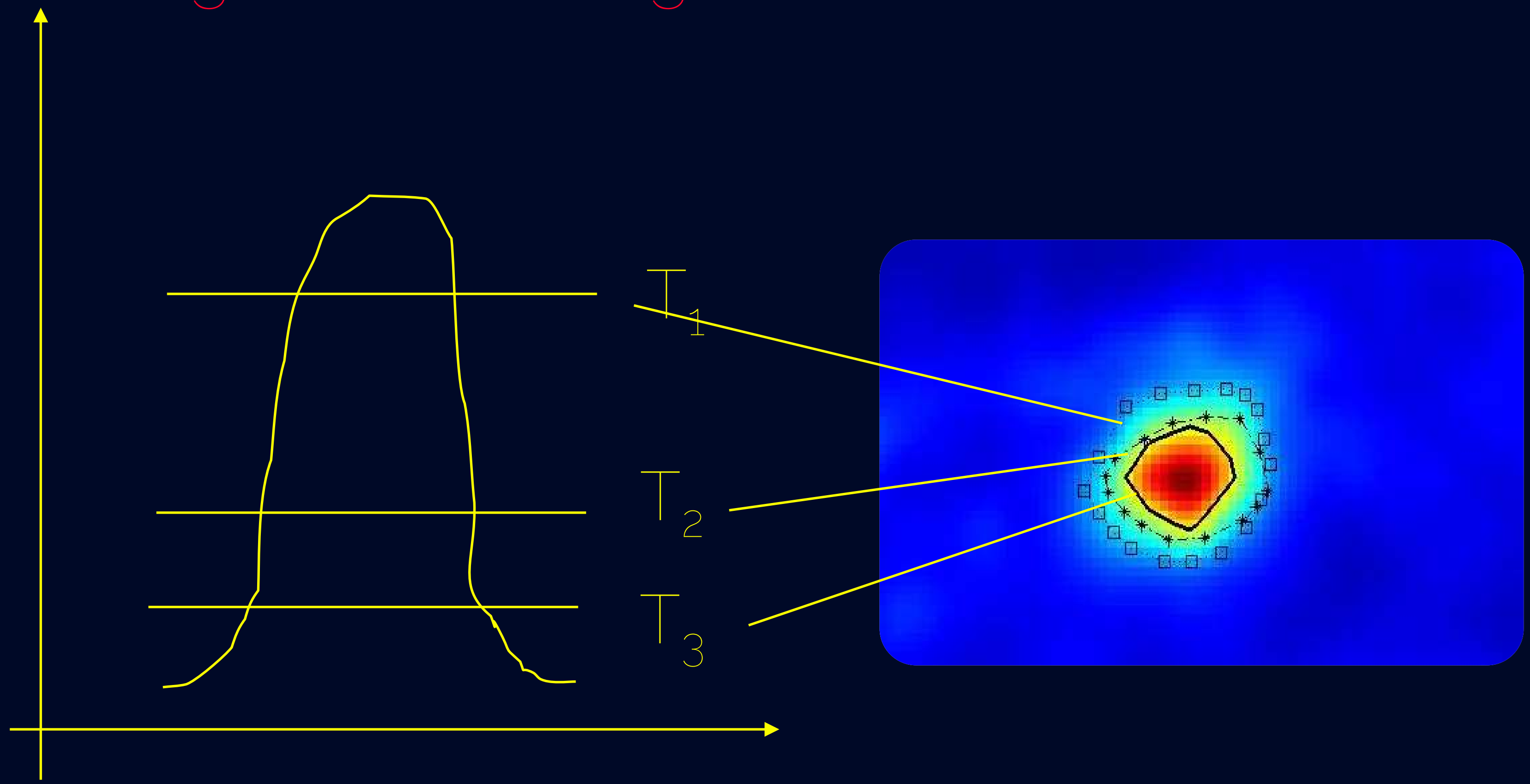


Input Image



threshold output

Image thresholding methods



$$ROI = T[I(x)] = \begin{cases} 1, & I(x) \geq T \\ 0, & I(x) < T \end{cases}$$

Image thresholding methods

with respect to lesion uptake (fixed)

%T = 40, 50, 60% of the maximum uptake

with respect to ROI volume

$$\%T = a + b \times \log_{10}(BTV) + q$$

a= 59.1 b = -18.5

a priori knowledge of the ROI volume

absolute

T = 2,5 cut off

PET/CT in oncology: quantitative analysis

SUV

Standardized Uptake Value

“Standardized”?

$$\text{SUV} = \frac{\text{Decay corrected dose/ ml of tumour}}{\text{Injected dose/ patient weight in gr}}$$

- body-weight corrected

$$SUV_{bw} = \frac{\text{Tissue concentration [kBq ml}^{-1}\text{]}}{\text{Injected dose [MBq]/ W[Kg]}}$$

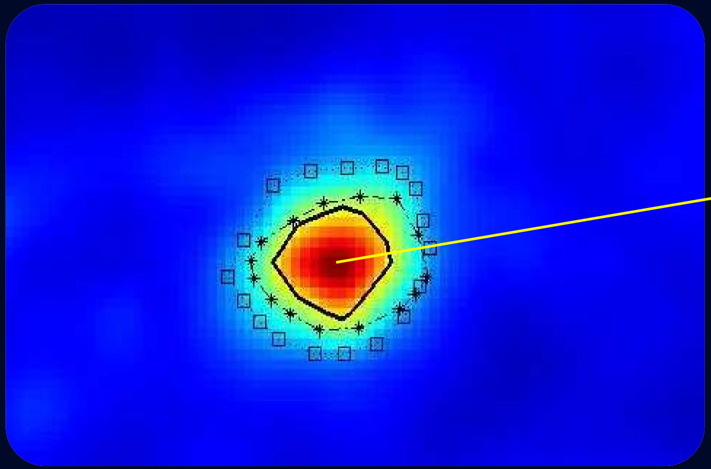
- body-surface-area corrected

$$SUV_{BSA} = \frac{\text{Tissue concentration [kBq ml}^{-1}\text{]}}{\text{Injected dose [MBq]/ } 0.007184 \times W^{0.425} \times h^{0.725} \text{[Kg]}}$$

- lean body-mass corrected

$$SUV_{LBM} = \frac{\text{Decay corrected dose/ ml of tumour}}{\text{Injected dose/ } 48.0 + 1.06 \times (h-152)}$$

Why SUV?



SUV=5.23

Metabolic response: EORTC Recommendations

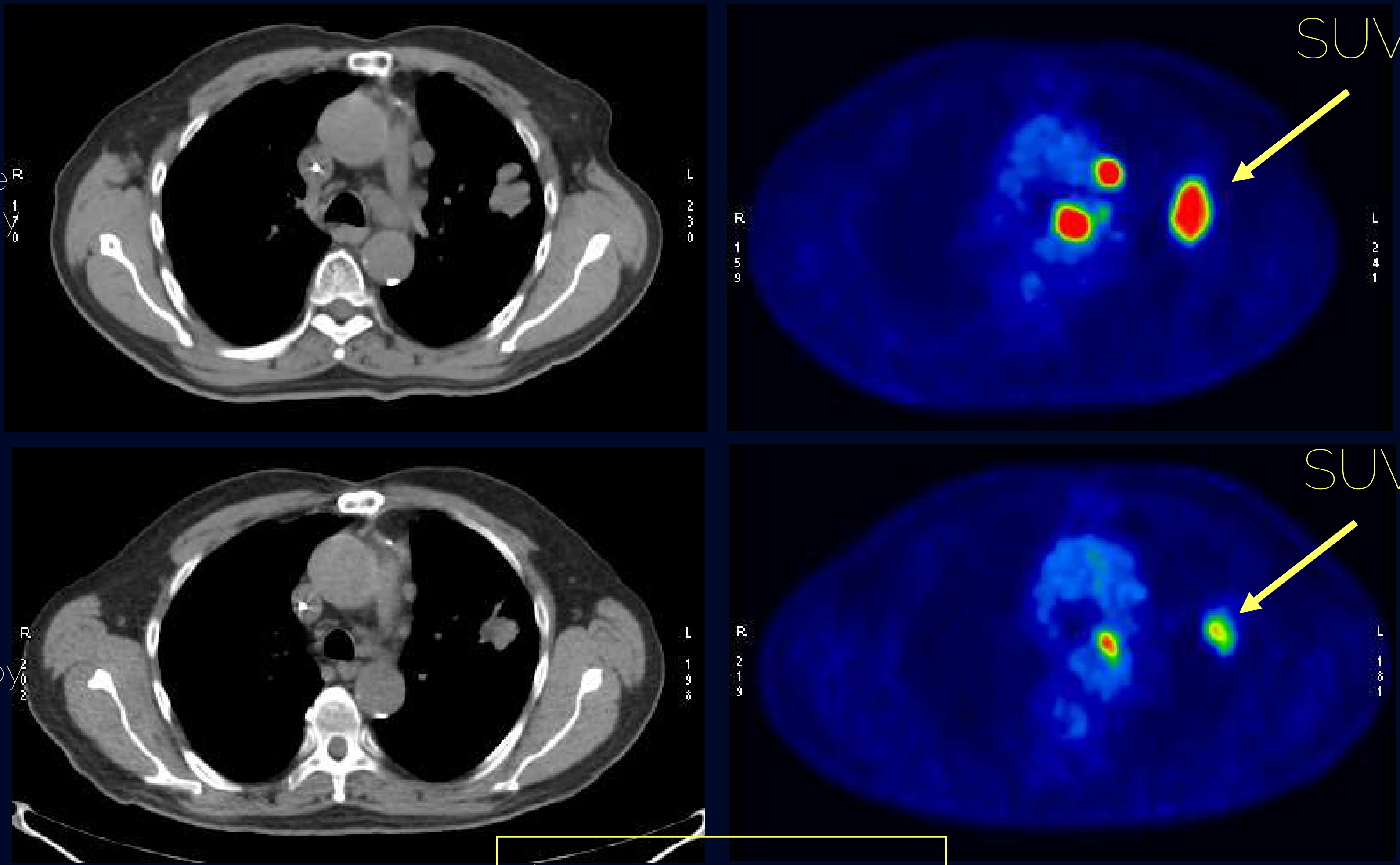
Progressive metabolic disease (PD)	**SUV > 25%↑** **Visual increase of extent, new locations**
Stable metabolic disease (SD)	**SUV < 25% ↑, < 15%↓** **No visible increase**
Partial metabolic response (PR)	**SUV > 15% ↓ one cycle CT** **SUV > 25% ↓ several cycles CT**
Complete metabolic response (CR)	**FDG uptake resolved**

Young et al., European Journal of Cancer 1999

PET/ CT Monitoring therapy responce

Before Therapy

After Therapy

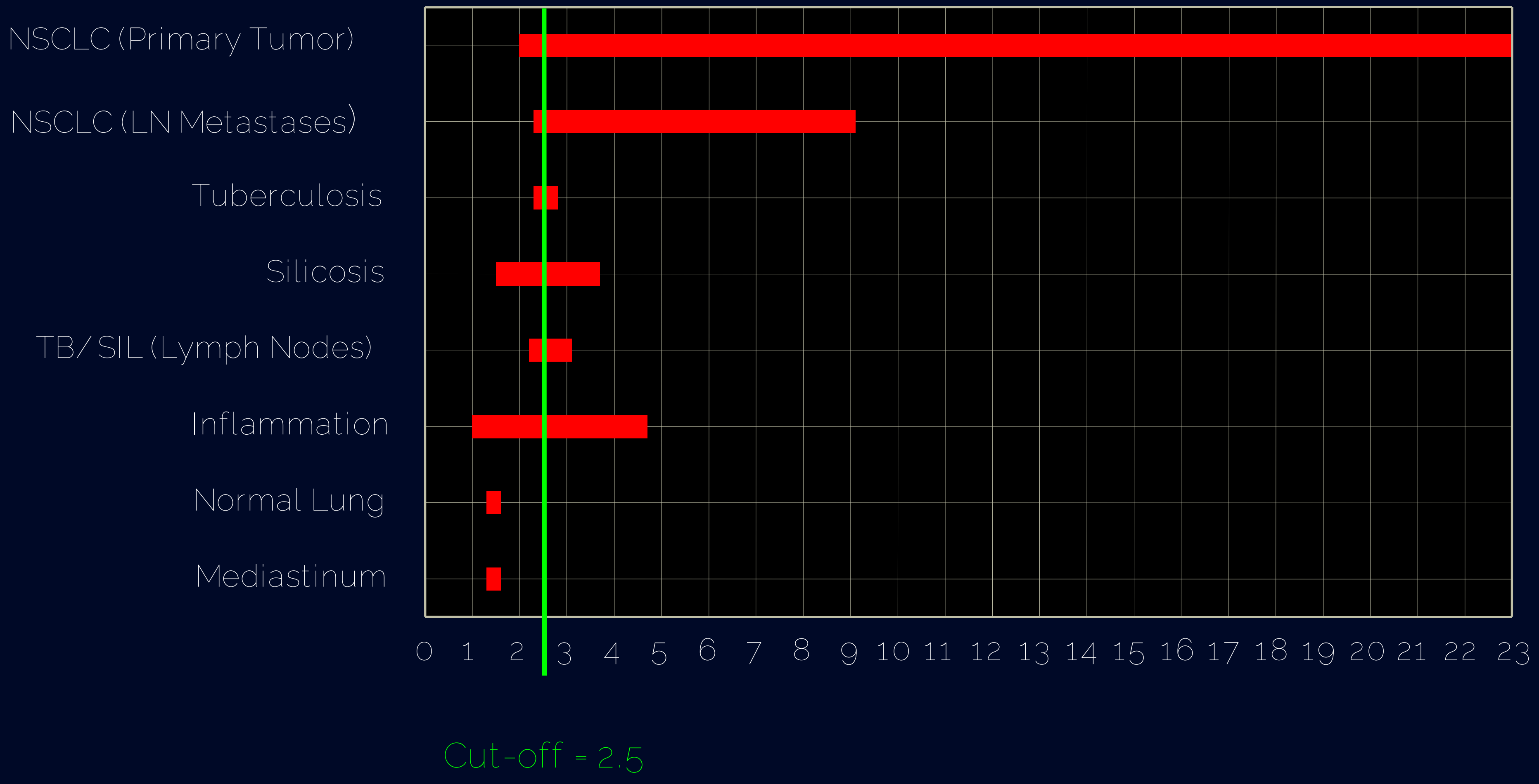


SUV = 8.9

SUV = 3.6

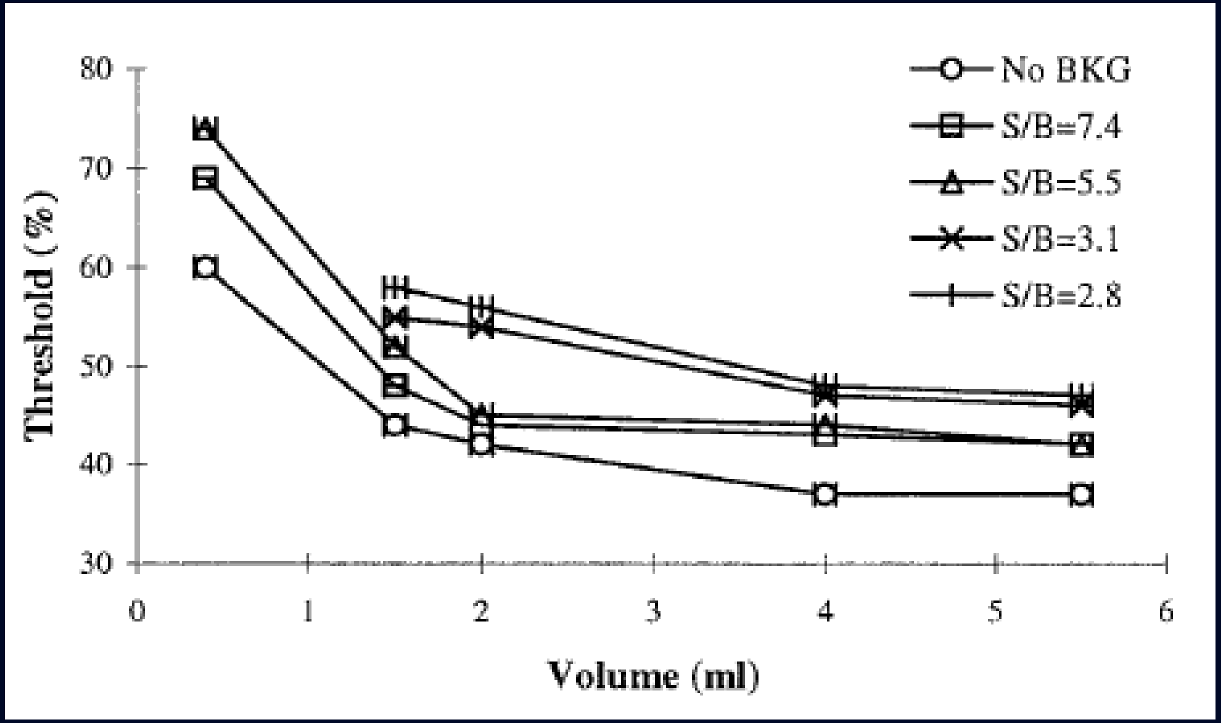
ΔSUV = - 59.5%

SUV in oncological and not-oncological diseases



with respect to lesion-to-background ratio (LB)

$T = a + b \times 1/LB$



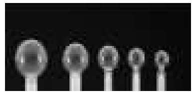
precalibration of LB vs lesion volume is necessary

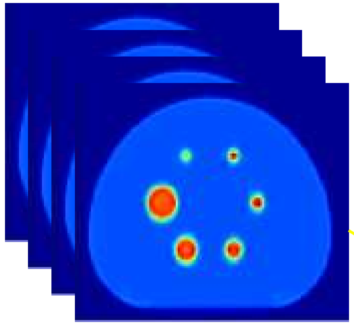
a priori knowledge of the lesion volume

Erdi et al.,1997

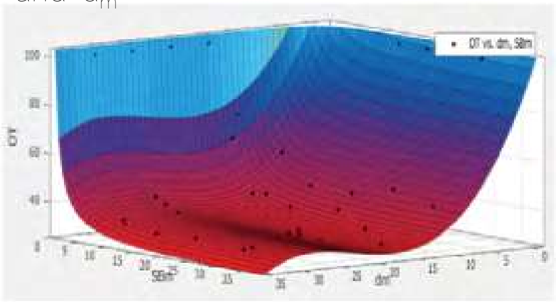
Solutions (Repeatable/Reproducible/Accurate)

NEMA IQ phantom





Adaptative Threshold = function of L/B_m and d_m



- Automatic segmentation of lesion volume

978-1-4799-0534-8/13/$31.00 ©2013 IEEE

Adaptive threshold method based on PET measured lesion-to-background ratio for the estimation of Metabolic Target Volume from [18]F-FDG PET images

Francesca Gallivanone, Federico Fazio, Luca Presotto, *Member, IEEE*, Maria C. Gilardi, Carla Canevari, Isabella Castiglioni

Hindawi Publishing Corporation
Computational and Mathematical Methods in Medicine

Research Article

An Adaptive Thresholding Method for BTV Estimation Incorporating PET Reconstruction Parameters: A Multicenter Study of the Robustness and the Reliability

M. Brambilla, R. Matheoud, C. Basile, C. Bracco, I. Castiglioni, C. Cavedon, M. Cremonesi, S. Morzenti, F. Fioroni, M. Giri, F. Botta, F. Gallivanone, E. Grassi, M. Pacilio, E. De Ponti, M. Stasi, S. Pasetto, S. Valzano, and D. Zanni

- Automatic Partial Volume correction

Hindawi Publishing Corporation
BioMed Research International
Volume 2013, Article ID 780458, 12 pages

Research Article

A Partial Volume Effect Correction Tailored for [18]F-FDG-PET Oncological Studies

F. Gallivanone,[1] C. Canevari,[2] L. Gianolli,[2] C. Salvatore,[3] P. A. Della Rosa,[1] M. C. Gilardi,[1] and I. Castiglioni[1]

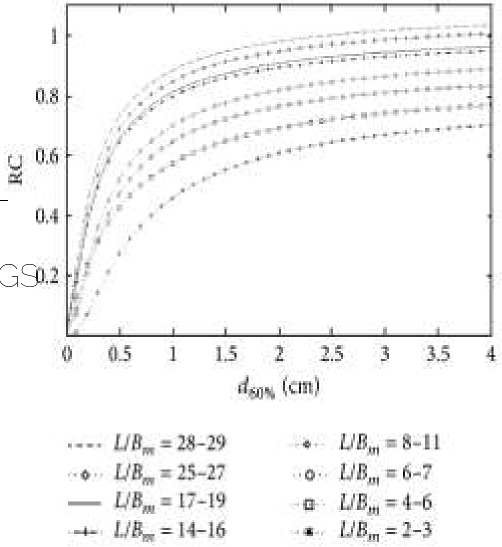
[1] IBFM-CNR, Via Elli Cervi 93, 20090 Segrate, Milan, Italy
[2] H San Raffaele, Via Olgettina 62, 20090 Segrate, Milan, Italy
[3] University of Milan-Bicocca, Milan, Italy

IEEE TRANSACTIONS ON NUCLEAR SCIENCE

PVE Correction in PET-CT Whole-Body Oncological Studies From PVE-Affected Images

Francesca Gallivanone, Alessandro Stefano, Eleonora Grosso, Carla Canevari, Luigi Gianolli, Cristina Messa, Maria Carla Gilardi, and Isabella Castiglioni

$$RC = \frac{L/B_m}{L/B_{GS}}$$



$$T = a + b \times SUV_{mean}$$

Black et al., 1990

$a = 0.588 \qquad b = 0.307$

$$T = \text{Contrast level} \times (L_{max} - B_{mean}) + B_{mean}$$

Drever et al., 2006

$$T = \beta \, L_{mean-70\%} + B_{mean}$$

Nestle et al., 2005

$\beta = 0.15$

$$T = (a \times SUV_{mean-70\%} + b \times B_{mean}) / SUV_{max}$$

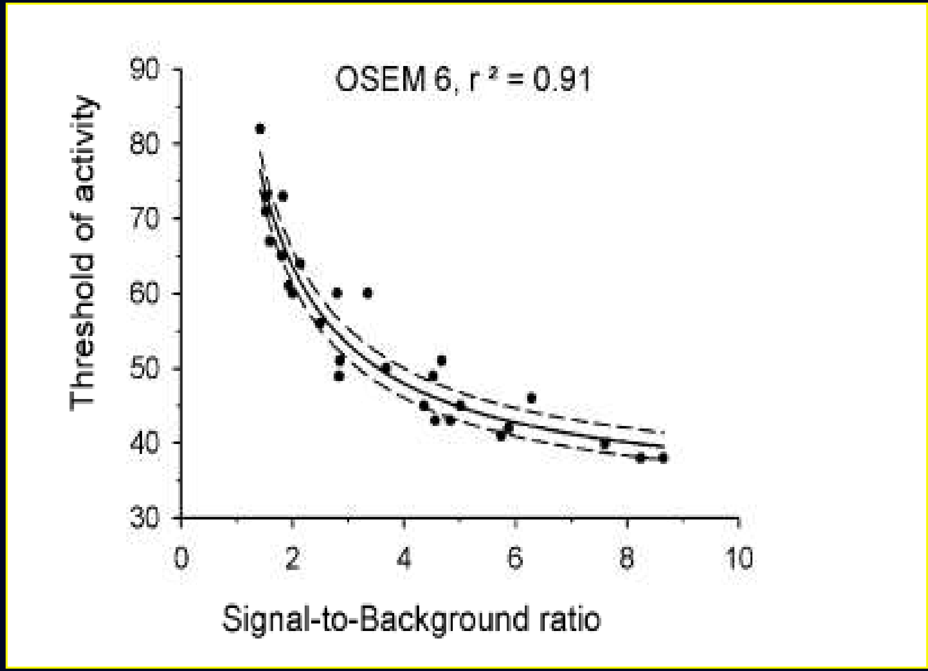
Schaefer et al., 2008

$a = 0.50$ $b = 0.50$ for $d > 3cm$

$a = 0.67$ $b = 0.60$ for $d < 3cm$ (Ecat ART)

%T = b x B/L + c

Daisne et al. , 2003



iterative

$$\%T = a/V_l + b \times B/L + c$$

Jentzen et al., 2007

$a = 7.8\%$ $b = 61.7\%$ $c = 31.6\%$

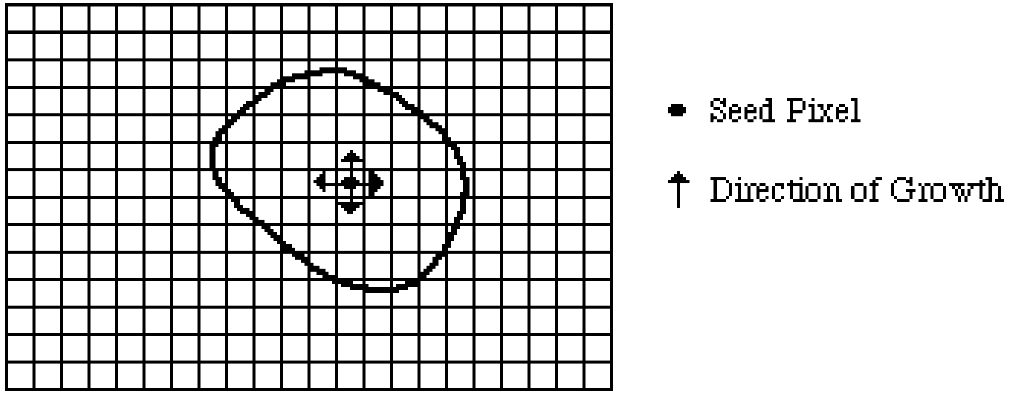
$$\%T = a_0 + \exp[a + b/V_l + c\log V_l]$$

Nehmeh et al., 2009

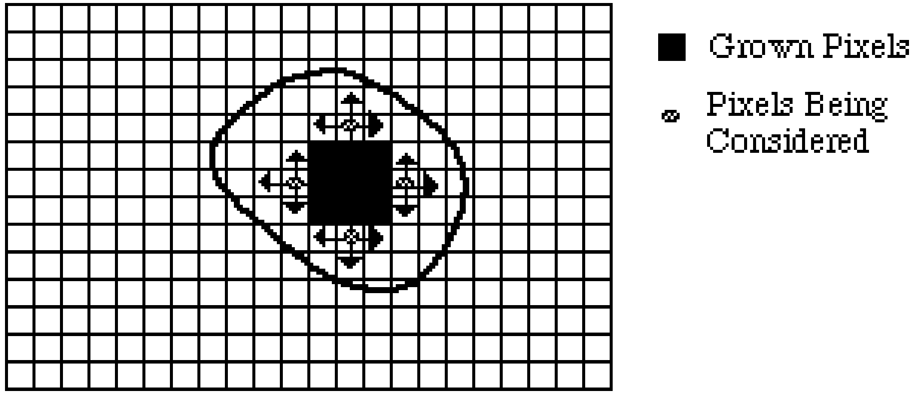
$a_0 = 5$ $a = 3.568$ $b = 0.197$ $c = -0.1069$

Image Segmentation

Region growing is a segmentation technique that divides an image into regions based on similarity criteria. It begins from a seed point and expands the region by incorporating neighboring pixels with similar characteristics.



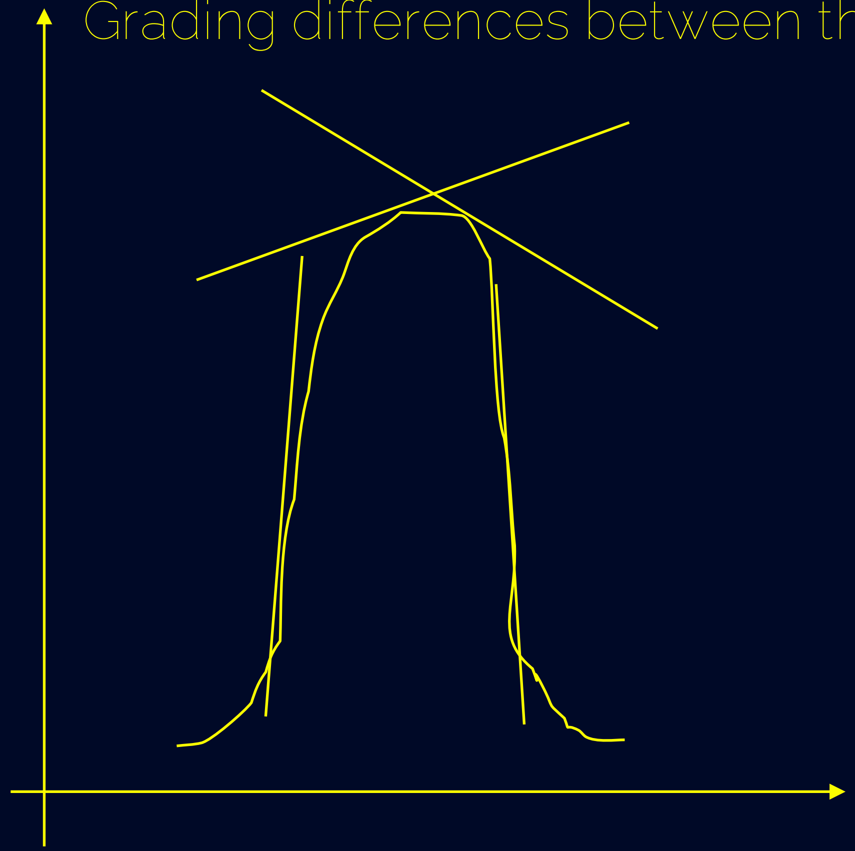
(a) Start of Growing a Region

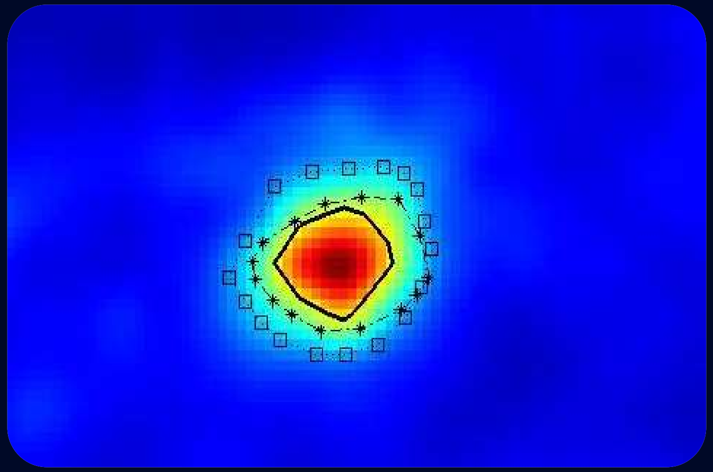


(b) Growing Process After a Few Iterations

(3) variational

Grading differences between the foreground lesion and B



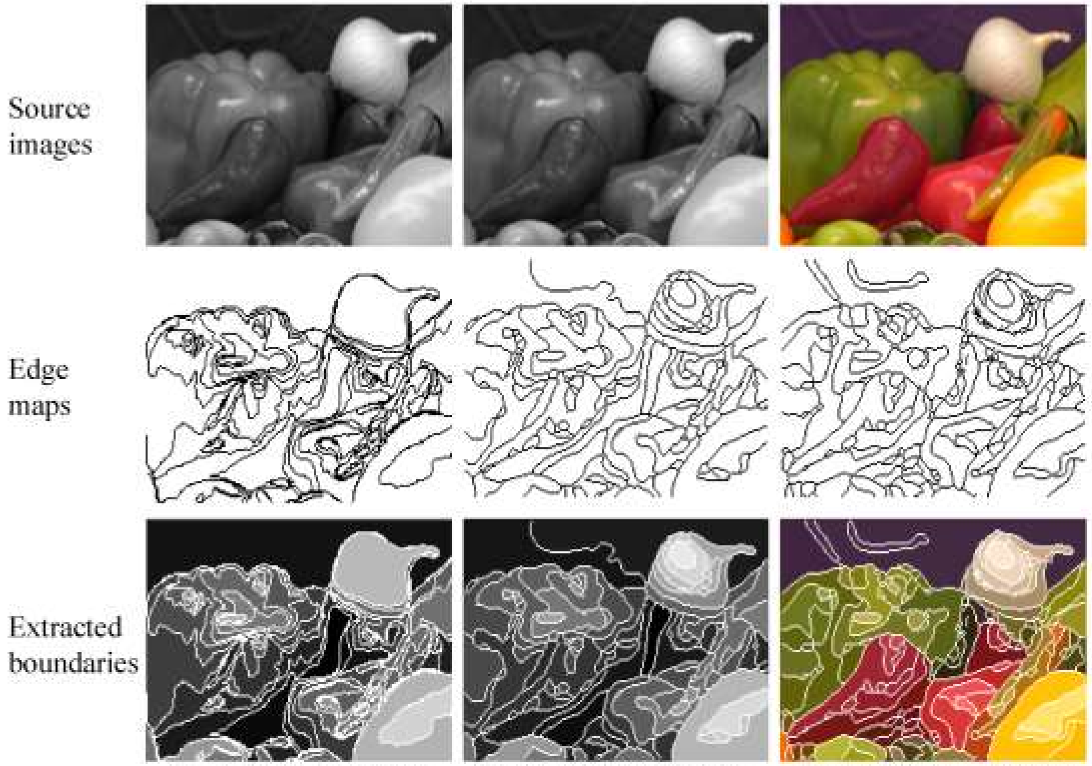


3a) edge detectors (Sobel operator)

3b) ridge detectors (Watershed Transform)

Image Segmentation

Edge-based segmentation relies on detecting edges in an image, which represent boundaries between different regions. This is achieved through edge detection algorithms.



(a) PMVIF with Gray (b) PMLCD with Gray (c) PMLCD with Color

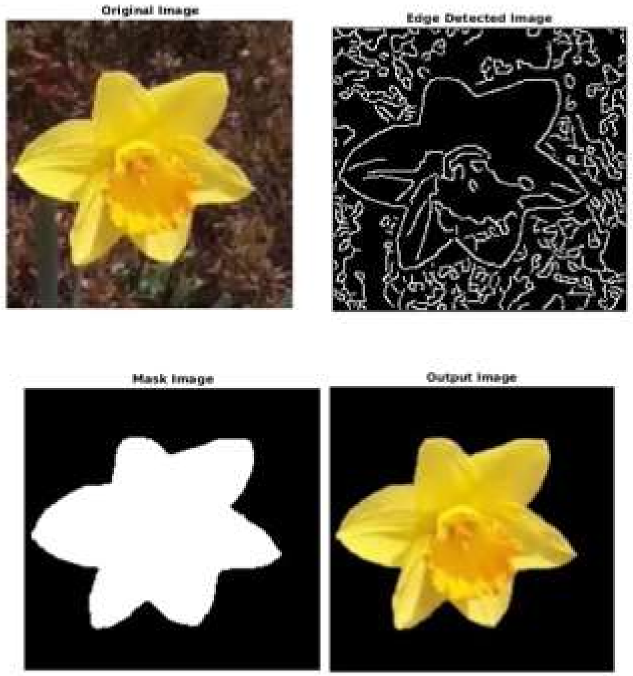


Image Segmentation

Watershed segmentation is based on the concept of flooding an image from its minima. The image is treated as a topographic relief, where intensity values indicate the height of the terrain.

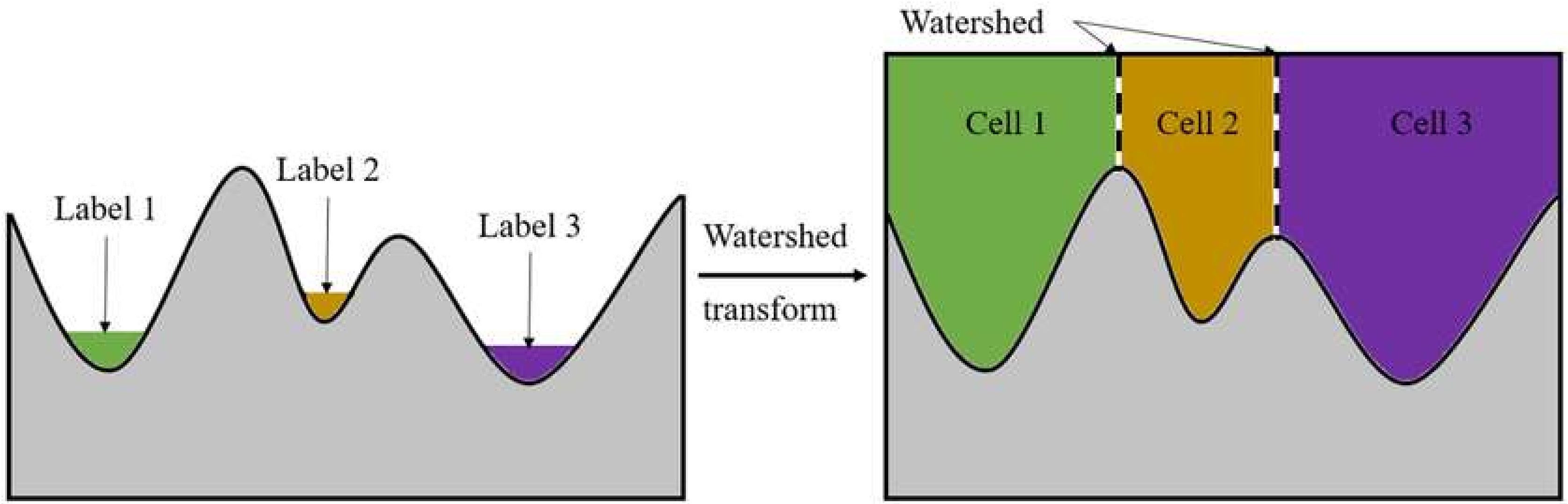
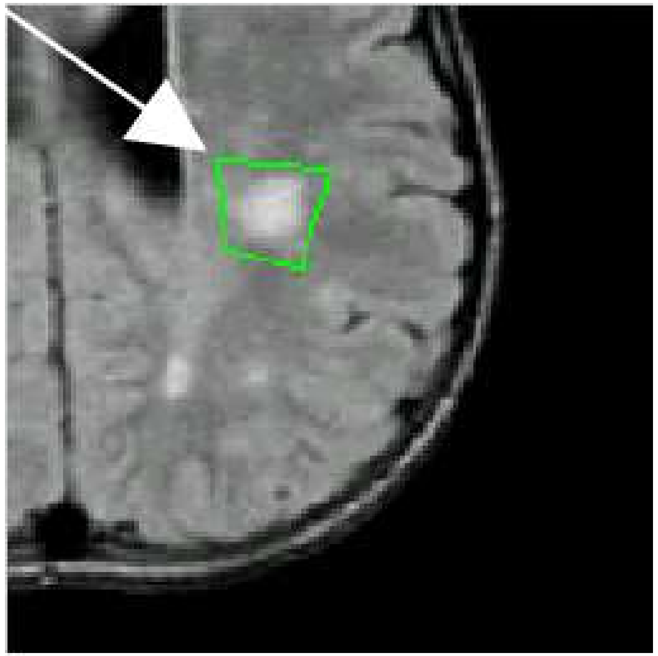


Image Segmentation

(Deformable) active contours, or snakes, are deformable curves employed to delineate object boundaries in an image. These curves are controlled by an energy function minimizing the distance between the curve and the object boundary.



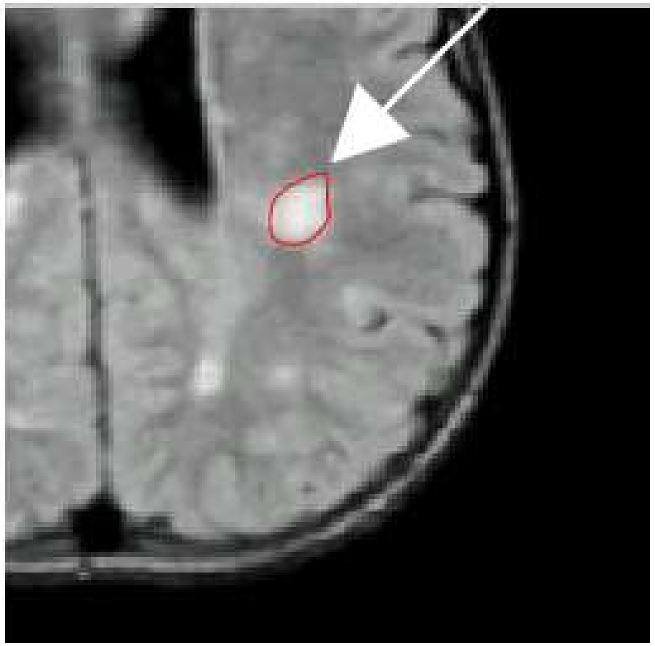


Image Segmentation

Graph-based segmentation represents an image as a graph and partitions it based on principles derived from graph theory.

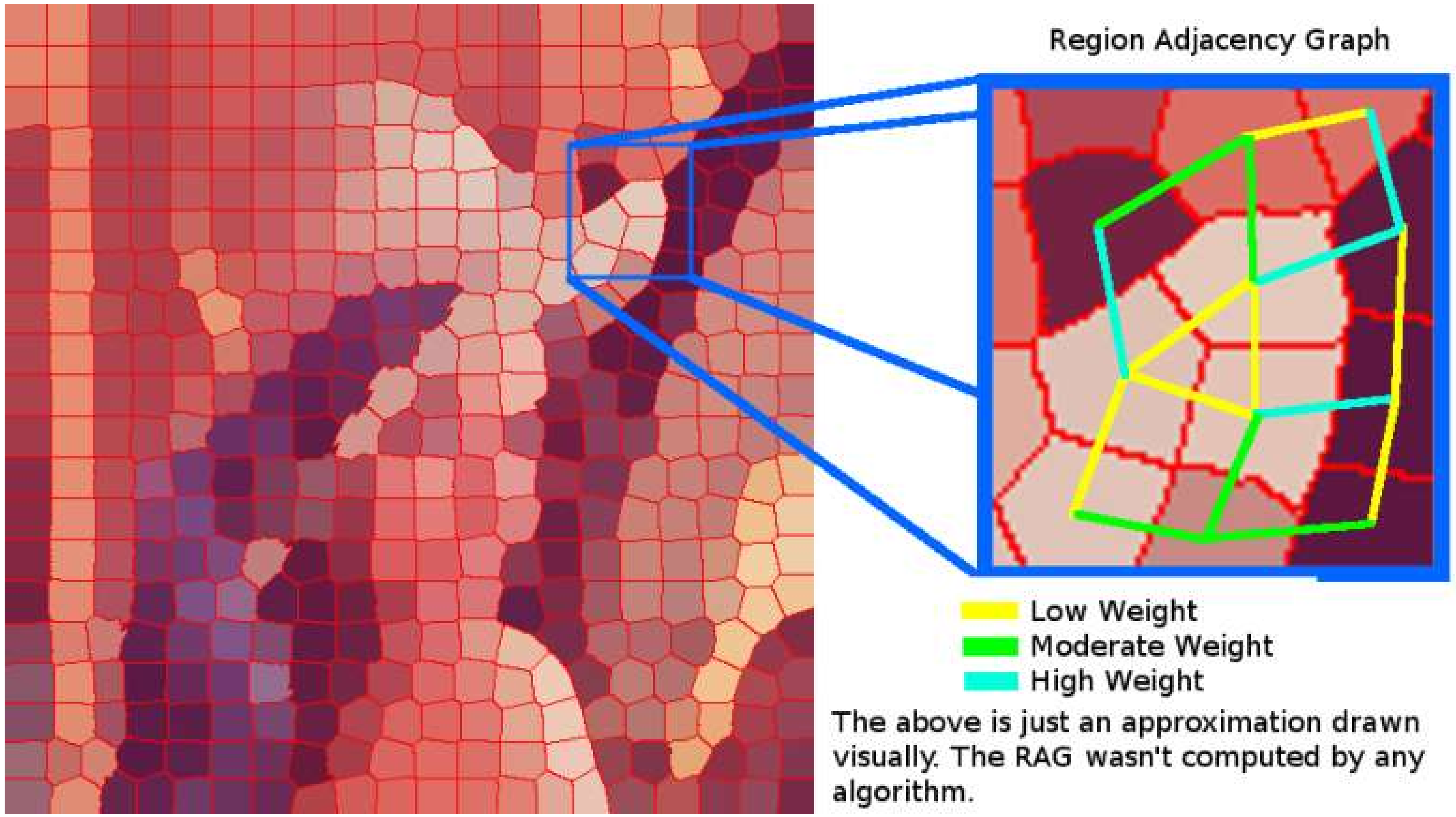


Image Segmentation

Superpixel-based segmentation groups similar image pixels together to create larger, more meaningful regions known as superpixels.

Clustering techniques group pixels into clusters according to similarity criteria, such as color, intensity, texture, or other features.

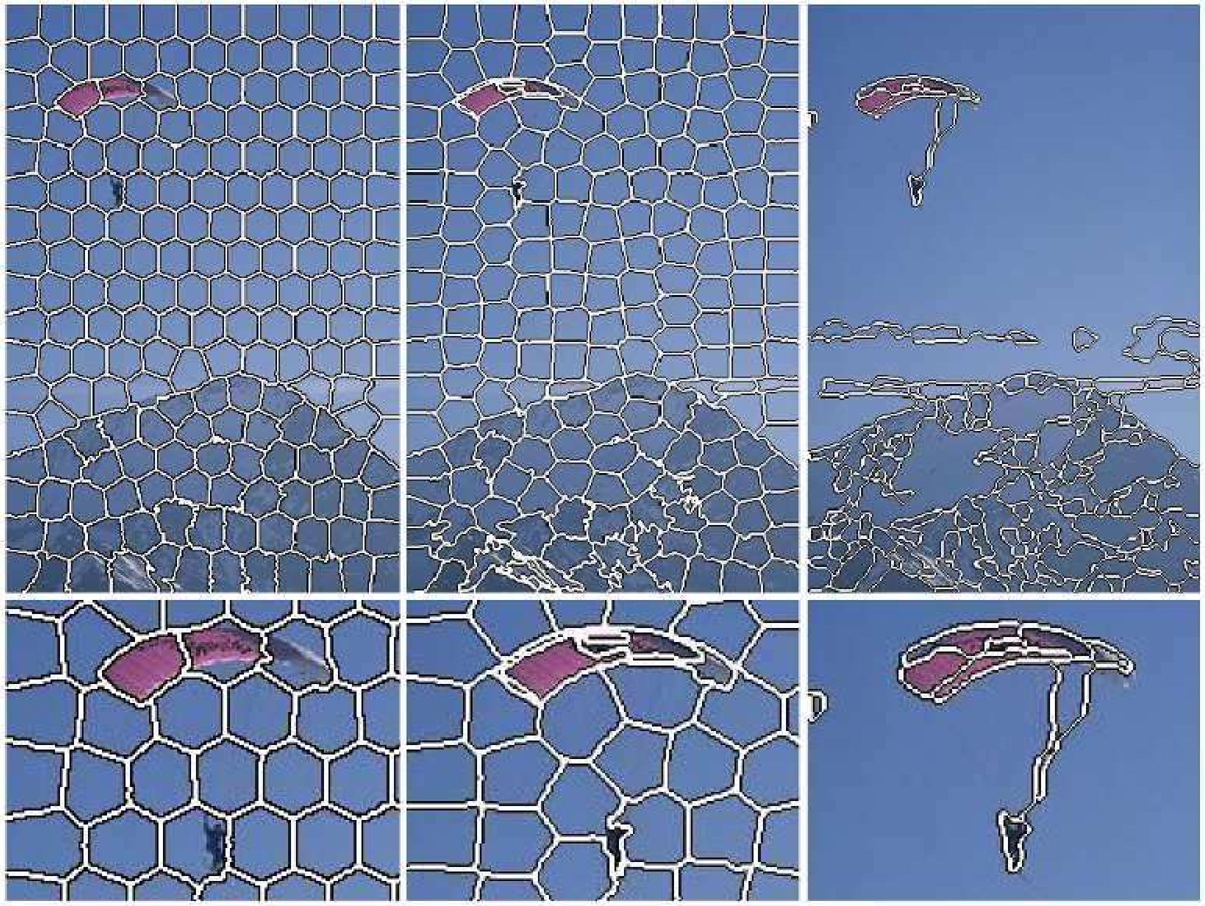
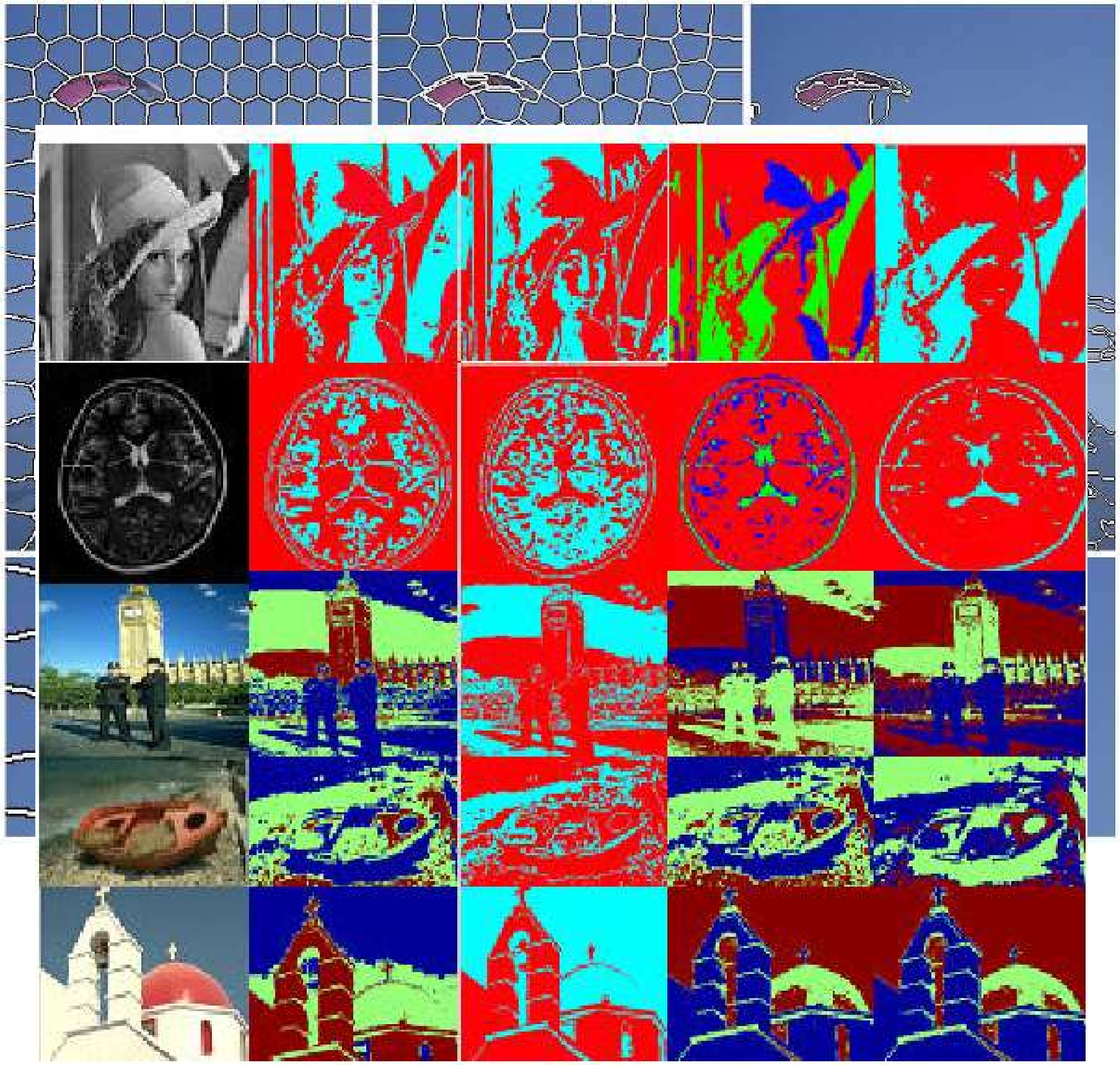


Image Segmentation

Superpixel-based segmentation groups similar image pixels together to create larger, more meaningful regions known as superpixels.



Clustering techniques group pixels into clusters according to similarity criteria, such as color, intensity, texture, or other features.

(4) learning methods

learning task aims to discriminate signals in the lesion voxels from surrounding normal tissue voxels based on a set of extracted features from these images

supervised

unsupervised

Supervised learning is used to estimate an unknown (input, output) mapping from known labelled samples called the training set (e.g. classification of lesions given a certain number of example images)

In unsupervised learning, only input samples are given to the learning system (e.g. clustering)

Classifiers:

k-nearest neighbour (KNN)

support vector machine (SVM)

artificial neural network (ANN)

Clustering:

k-means algorithm

fuzzy C-means (FCM) algorithm

Expectation maximization (EM) algorithm

Image Segmentation

Deep learning-based segmentation, particularly using Convolutional Neural Networks (CNNs), has significantly advanced image segmentation. These techniques employ a hierarchical approach, applying multiple layers of filters to extract high-level features from the input image.

U-Net is a convolutional neural network (CNN) architecture designed for semantic segmentation tasks, particularly in medical image analysis. It was introduced by Ronneberger et al. in 2015

Key Components:

U-shaped Architecture:

1. UNet is characterized by its distinctive U-shaped architecture. It consists of a contracting path (encoder) and an expansive path (decoder), forming a U-shaped network.

Contracting Path (Encoder):

1. The contracting path captures hierarchical features through a series of convolutional and pooling layers. This helps in learning and extracting abstract representations from the input image, gradually reducing spatial dimensions.

1. Contracting path (encoder)
2. Expansive path (decoder)

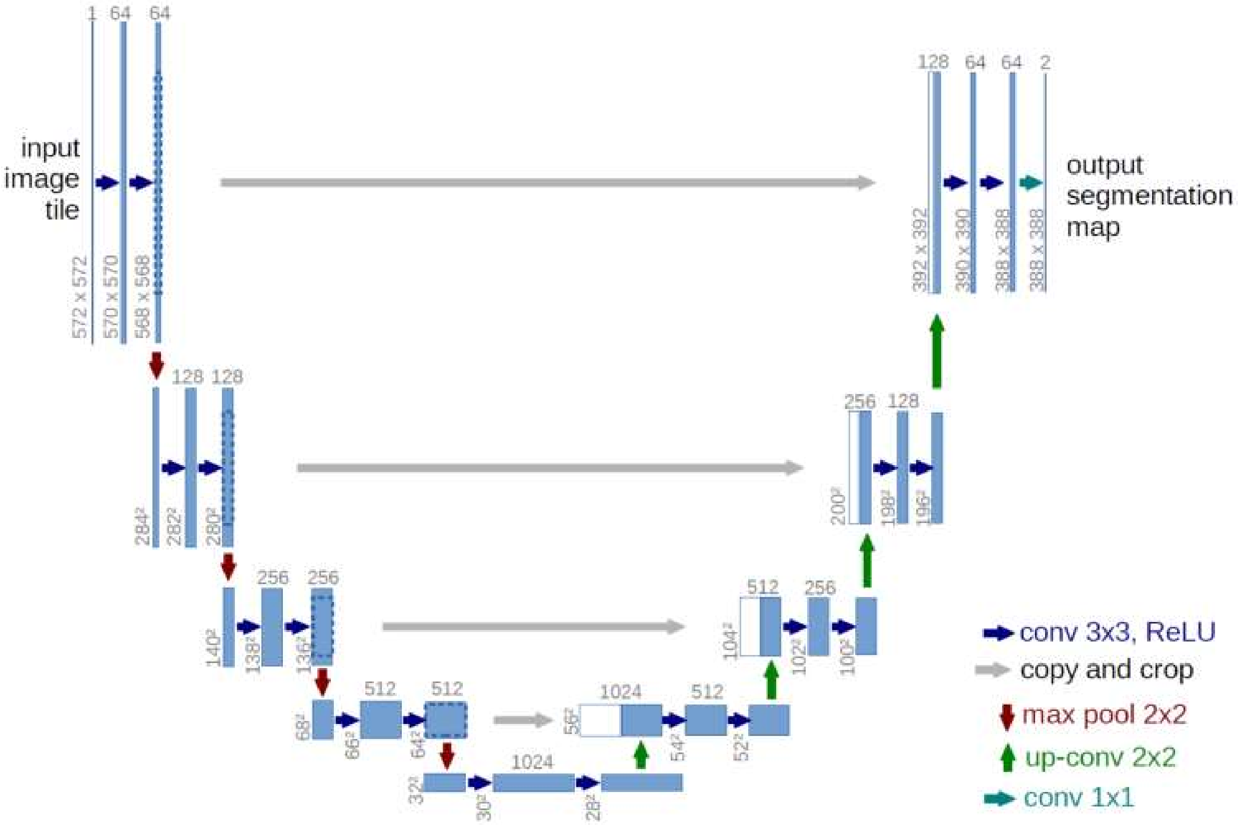


Figure 2.25: Original U-net architecture, as presented in [61]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Key Components:

Expansive Path (Decoder):

1. The expansive path reconstructs the segmented image by employing upsampling and concatenation operations. This path gradually increases spatial resolution and refines the segmentation details.

Skip Connections:

1. UNet incorporates skip connections between corresponding encoder and decoder layers. These connections allow the decoder to access high-resolution features from the encoder, aiding in the preservation of spatial information.

1. Contracting path (encoder)
2. Expansive path (decoder)

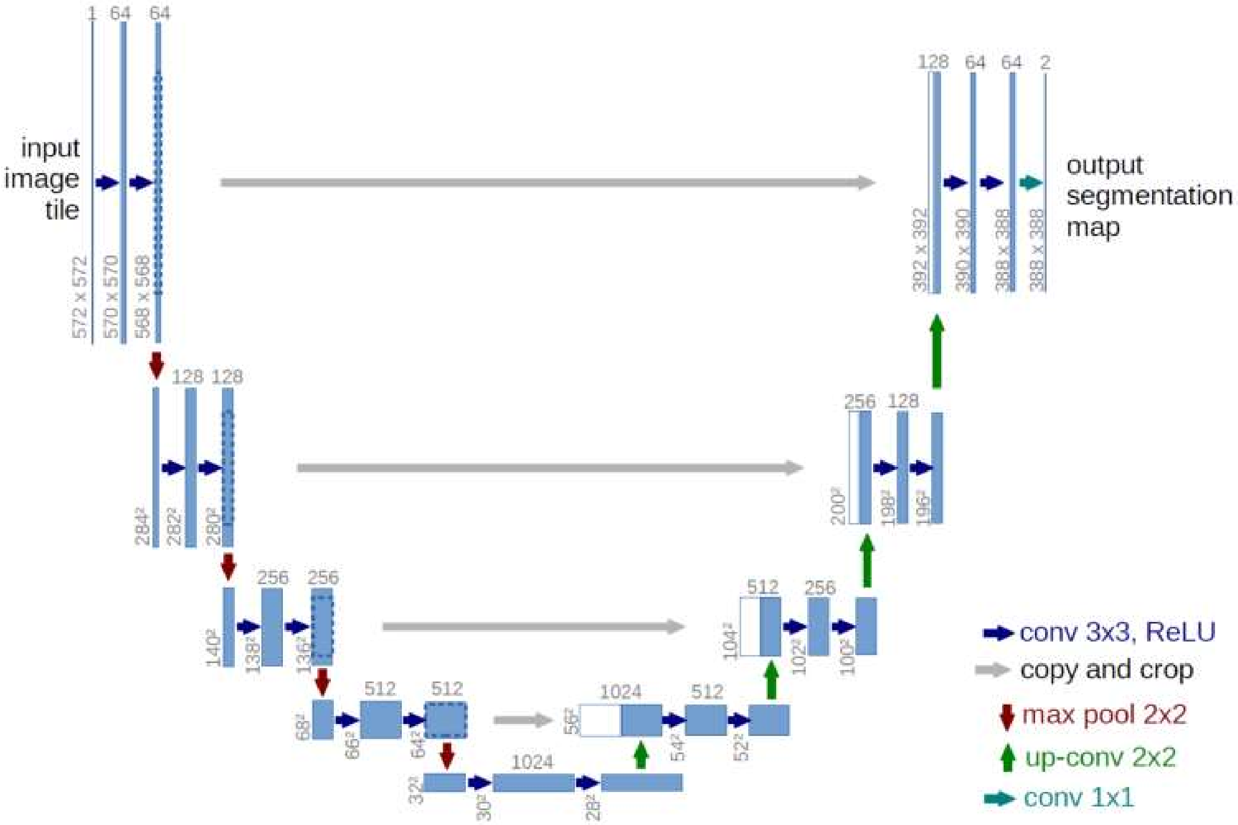


Figure 2.25: Original U-net architecture, as presented in [61]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Image Segmentation

1. Contracting path (encoder)
2. Expansive path (decoder)

Key Components:

Convolutional Blocks:

1. Both encoder and decoder consist of convolutional blocks, typically using rectified linear units (ReLU) as activation functions. Batch normalization is often employed to stabilize and accelerate training.

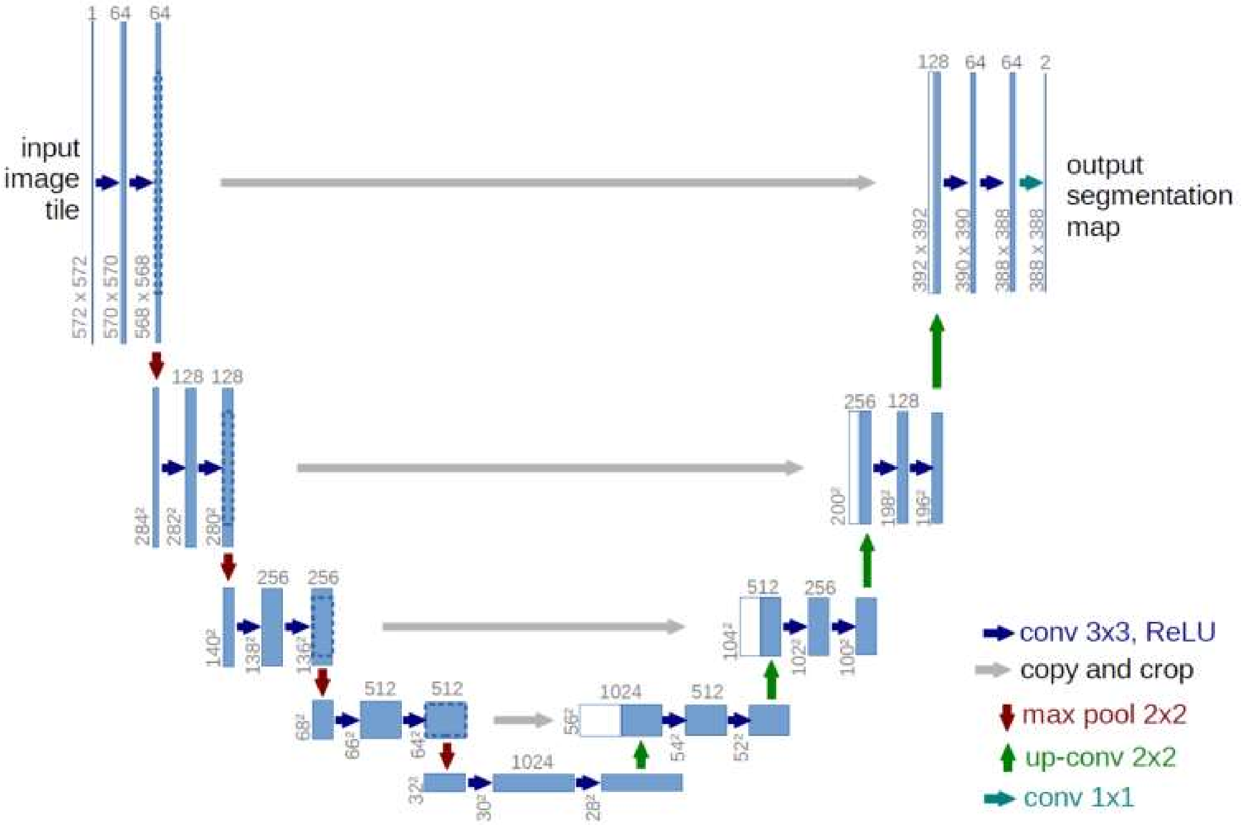


Figure 2.25: Original U-net architecture, as presented in [61]. Each blue box corresponds to a multi-channel feature map. The number of channels is denoted on top of the box. The x-y-size is provided at the lower left edge of the box. White boxes represent copied feature maps. The arrows denote the different operations.

Image Segmentation

Scientific Rationale for Image Segmentation:

Multi-Scale Feature Learning:

1. The contracting path captures features at multiple scales, facilitating the learning of both low-level and high-level representations. This is crucial for recognizing fine details and global context in images.

Contextual Information Preservation:

1. Skip connections enable the integration of contextual information from different scales, aiding in the preservation of spatial details during the upsampling process in the decoder.

Effective Feature Combination:

1. The concatenation of feature maps from the encoder with those from the decoder allows for effective feature combination, ensuring that the decoder has access to relevant high-resolution information for accurate segmentation.

Adaptability to Various Image Sizes:

1. UNet's architecture allows it to handle input images of various sizes, making it versatile for different segmentation tasks without requiring fixed input dimensions.

Reducing the Need for Massive Amounts of Labeled Data:

1. Due to its architecture, UNet has shown effectiveness in scenarios with limited labeled training data. The skip connections help propagate information from the encoded features, enabling better generalization.

Image Segmentation

```
Model: "model"
__________________________________________________________________________________________________
 Layer (type)                   Output Shape         Param #     Connected to
==================================================================================================
 input_1 (InputLayer)           [(None, 128, 128, 1)]  0         []

 lambda (Lambda)                (None, 128, 128, 1)  0           ['input_1[0][0]']

 conv2d (Conv2D)                (None, 128, 128, 16)  160        ['lambda[0][0]']

 dropout (Dropout)              (None, 128, 128, 16)  0          ['conv2d[0][0]']

 conv2d_1 (Conv2D)              (None, 128, 128, 16)  2320       ['dropout[0][0]']

 max_pooling2d (MaxPooling2     (None, 64, 64, 16)   0           ['conv2d_1[0][0]']
 D)

 conv2d_2 (Conv2D)              (None, 64, 64, 32)   4640        ['max_pooling2d[0][0]']

 dropout_1 (Dropout)            (None, 64, 64, 32)   0           ['conv2d_2[0][0]']

 conv2d_3 (Conv2D)              (None, 64, 64, 32)   9248        ['dropout_1[0][0]']

 max_pooling2d_1 (MaxPoolin     (None, 32, 32, 32)   0           ['conv2d_3[0][0]']
 g2D)

 conv2d_4 (Conv2D)              (None, 32, 32, 64)   18496       ['max_pooling2d_1[0][0]']

 dropout_2 (Dropout)            (None, 32, 32, 64)   0           ['conv2d_4[0][0]']

 conv2d_5 (Conv2D)              (None, 32, 32, 64)   36928       ['dropout_2[0][0]']

 max_pooling2d_2 (MaxPoolin     (None, 16, 16, 64)   0           ['conv2d_5[0][0]']
 g2D)

 conv2d_6 (Conv2D)              (None, 16, 16, 128)  73856       ['max_pooling2d_2[0][0]']

 dropout_3 (Dropout)            (None, 16, 16, 128)  0           ['conv2d_6[0][0]']

 conv2d_7 (Conv2D)              (None, 16, 16, 128)  147584      ['dropout_3[0][0]']
```

[...]

```
 conv2d_13 (Conv2D)             (None, 32, 32, 64)   36928       ['dropout_6[0][0]']

 conv2d_transpose_2 (Conv2D     (None, 64, 64, 32)   8224        ['conv2d_13[0][0]']
 Transpose)

 concatenate_2 (Concatenate     (None, 64, 64, 64)   0           ['conv2d_transpose_2[0][0]',
 )                                                                'conv2d_3[0][0]']

 conv2d_14 (Conv2D)             (None, 64, 64, 32)   18464       ['concatenate_2[0][0]']

 dropout_7 (Dropout)            (None, 64, 64, 32)   0           ['conv2d_14[0][0]']

 conv2d_15 (Conv2D)             (None, 64, 64, 32)   9248        ['dropout_7[0][0]']

 conv2d_transpose_3 (Conv2D     (None, 128, 128, 16)  2064       ['conv2d_15[0][0]']
 Transpose)

 concatenate_3 (Concatenate     (None, 128, 128, 32)  0          ['conv2d_transpose_3[0][0]',
 )                                                                'conv2d_1[0][0]']

 conv2d_16 (Conv2D)             (None, 128, 128, 16)  4624       ['concatenate_3[0][0]']

 dropout_8 (Dropout)            (None, 128, 128, 16)  0          ['conv2d_16[0][0]']

 conv2d_17 (Conv2D)             (None, 128, 128, 16)  2320       ['dropout_8[0][0]']

 conv2d_18 (Conv2D)             (None, 128, 128, 1)  17          ['conv2d_17[0][0]']

==================================================================================================
Total params: 1940817 (7.40 MB)
Trainable params: 1940817 (7.40 MB)
Non-trainable params: 0 (0.00 Byte)
```

Image Segmentation Metrics

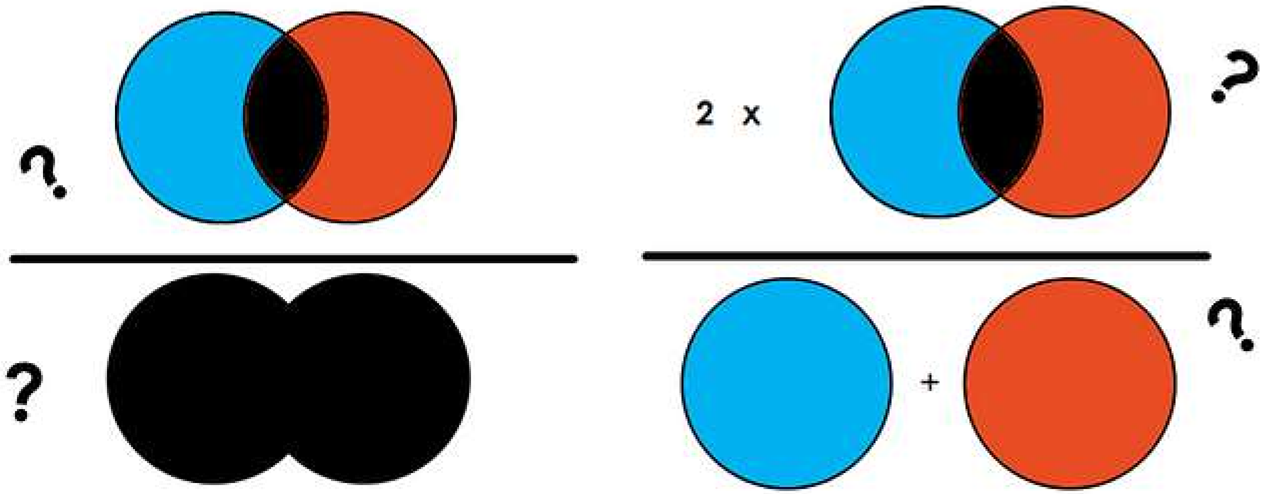


Image Segmentation Metrics

Accuracy:

1. Formula: Accuracy = (True Positives + True Negatives) / Total Pixels
2. Measures the overall correctness of pixel-wise segmentation. It is sensitive to class imbalance.

Precision:

1. Formula: Precision = True Positives / (True Positives + False Positives)
2. Measures the accuracy of positive predictions. High precision indicates a low rate of false positives.

Recall (Sensitivity):

1. Formula: Recall = True Positives / (True Positives + False Negatives)
2. Measures the ability to correctly identify positive instances. High recall indicates a low rate of false negatives.

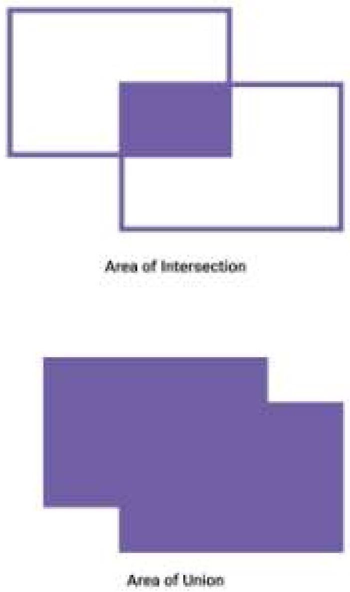
F1 Score:

1. Formula: F1 Score = 2 * (Precision * Recall) / (Precision + Recall)
2. Harmonic mean of precision and recall. It provides a balanced measure between precision and recall.

Image Segmentation Metrics

IoU (Intersection over Union) or Jaccard Index:

1. Formula: IoU = (Intersection of Predicted and True Segmentation) / (Union of Predicted and True Segmentation)
2. Measures the overlap between the predicted segmentation and the ground truth. A higher IoU indicates better segmentation accuracy.



$$\text{IOU} = \frac{\text{Area of Intersection}}{\text{Area of Union}}$$

$$Jaccard(A, B) = \frac{||A \cap B||}{||A \cup B||}$$

Mean Intersection over Union (mIoU):

1. Formula: mIoU = Average of IoU for all classes
2. Computes the average IoU across all classes, providing a global measure of segmentation accuracy.

Image Segmentation Metrics

Dice Coefficient:

1. Formula: Dice = 2 * (Intersection of Predicted and True Segmentation) / (Sum of Predicted and True Segmentation)
2. Similar to IoU, Dice coefficient measures the similarity between the predicted and true segmentation masks. It ranges from 0 (no overlap) to 1 (perfect overlap).

$$Dice(A, B) = \frac{2\|A \cap B\|}{\|A\| + \|B\|}$$

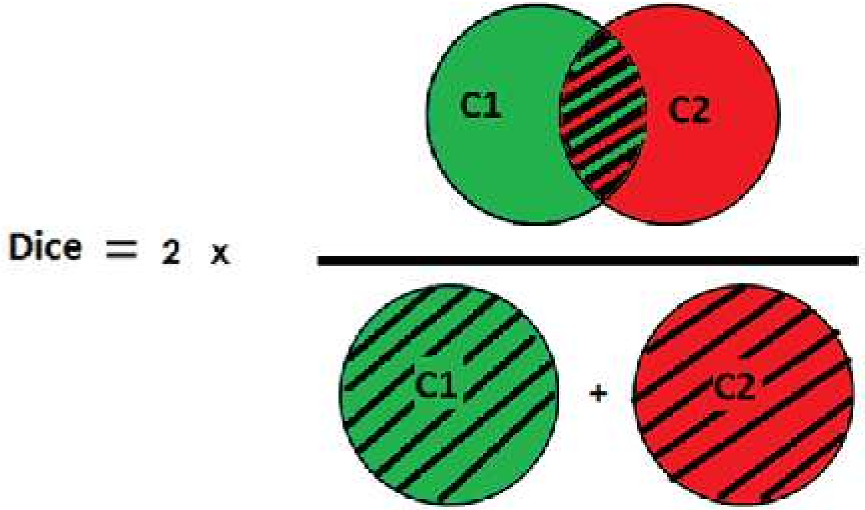


Image Segmentation Metrics

Surface Dice Overlap:

1. Evaluates the accuracy of 3D image segmentation by measuring the overlap of segmented surfaces.

Hausdorff Distance:

1. Measures the maximum distance between the predicted and true segmentation boundaries. It is sensitive to outliers.

Boundary F1 Score:

1. Focuses on the quality of boundary delineation in segmentation masks.