

# Advanced Deep Learning for Multi-View Structural Reasoning in Mammographic Analysis

## Internship Overview

Student : Imade Bouftini  
Supervision : Youssef ALJ

October 20, 2025

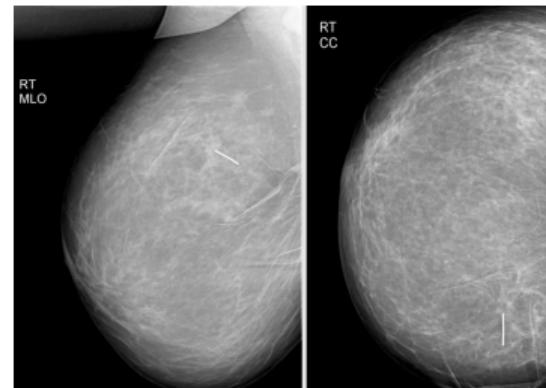
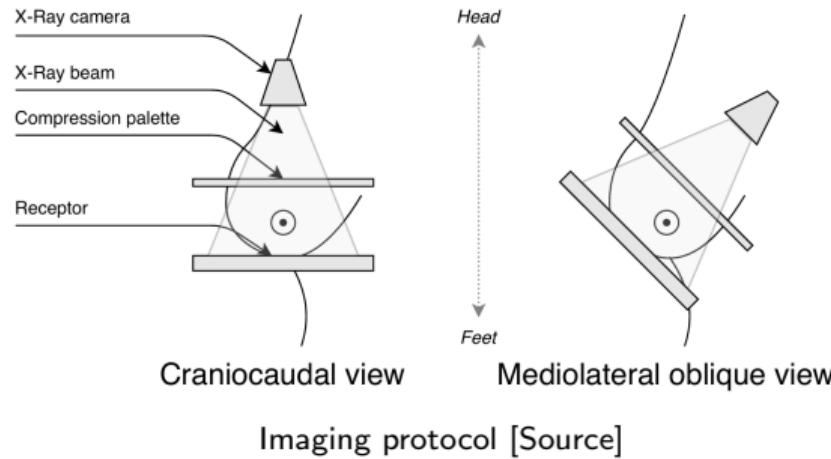
1. Clinical Context & Problem Statement
2. Single-view detection
3. Multi-view detection
4. Results & comparison
5. Conclusion & Perspectives

Section I

# Introduction

# Clinical Context: Breast Cancer Screening Global Impact and Detection Methodology

- Breast cancer affects millions worldwide (2.3M new cases annually)
- Early detection can reduce mortality by 20-40%
- Mammography is the primary screening tool



Mammogram Views (MLO/CC)

# Clinical Context: Detection Challenges

## Balancing Sensitivity and Specificity in Screening

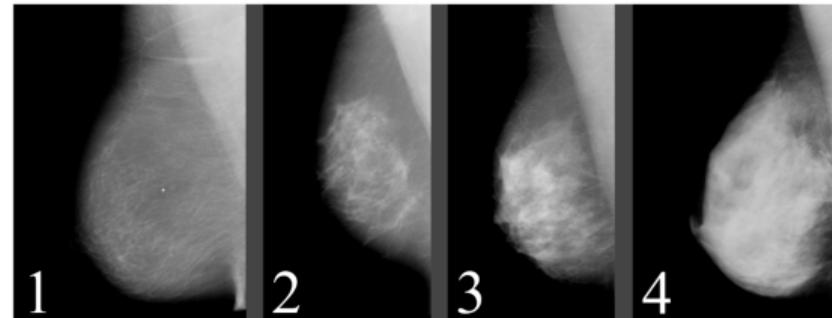


Figure: Mammograms with various density levels [Source]

### Clinical Challenges:

- Diverse tissue density and appearance variation between patients
- Extremely low prevalence ( $\sim 0.5\%$  in screening populations)
- Subtle presentation of early-stage cancers

#### False negatives cause:

- Delayed diagnosis
- Poorer prognosis
- Increased treatment costs

#### False positives cause:

- Unnecessary biopsies
- Patient anxiety and stress
- Healthcare resource burden

# Clinical Context: Multi-View Integration Mimicking Radiologist Reasoning with Multiple Views

## Core Challenge

How can we leverage ipsilateral and bilateral views to replicate the natural reasoning ability of radiologists in breast cancer detection?

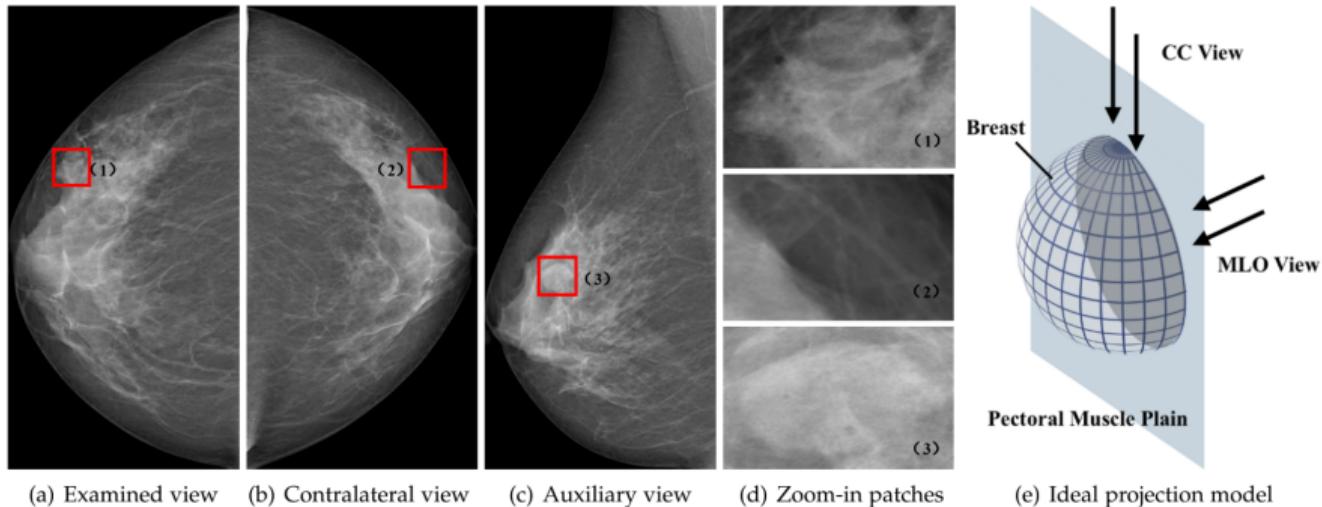
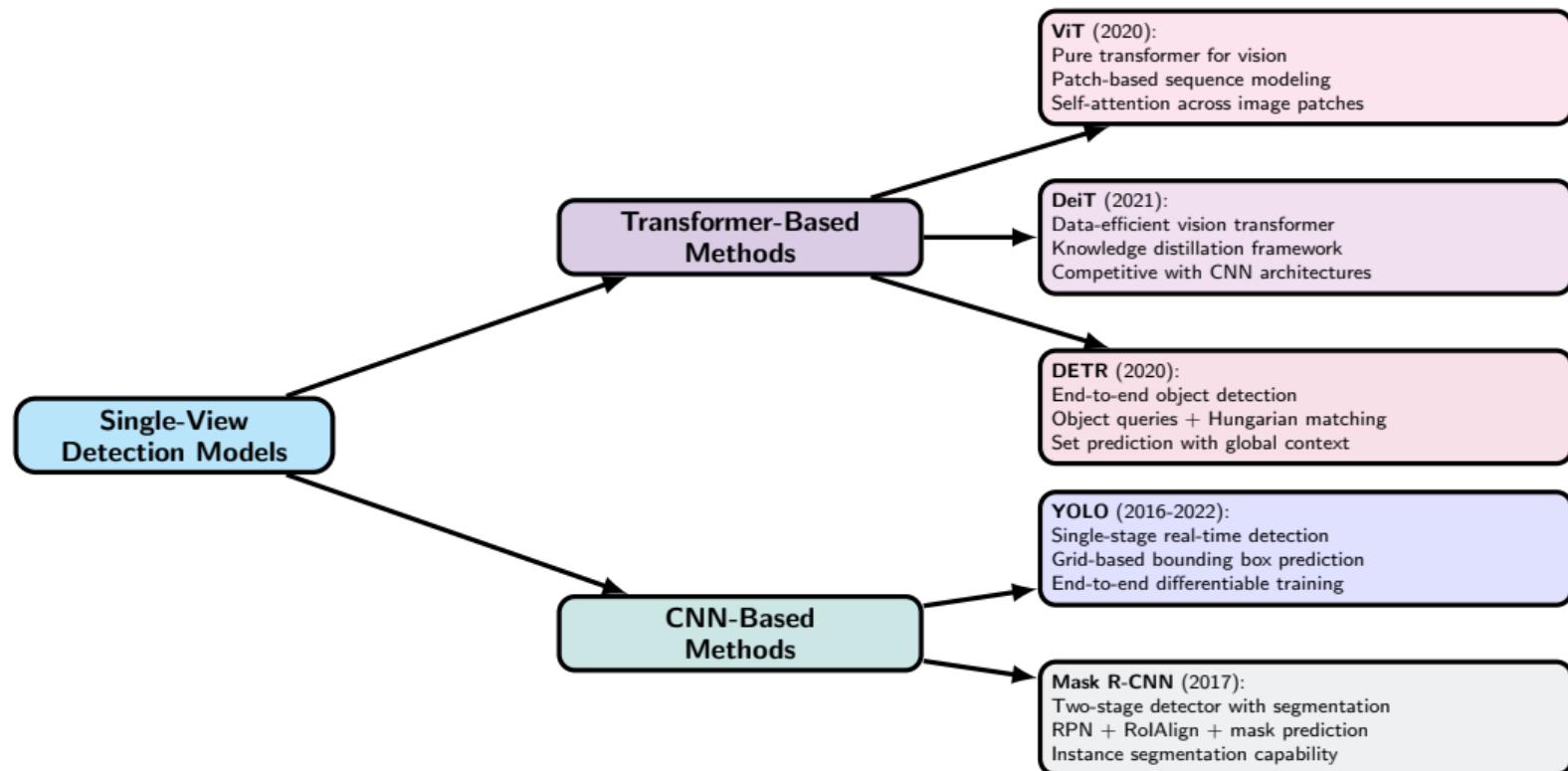


Figure: Illustration of the relation among mammography views [Source]

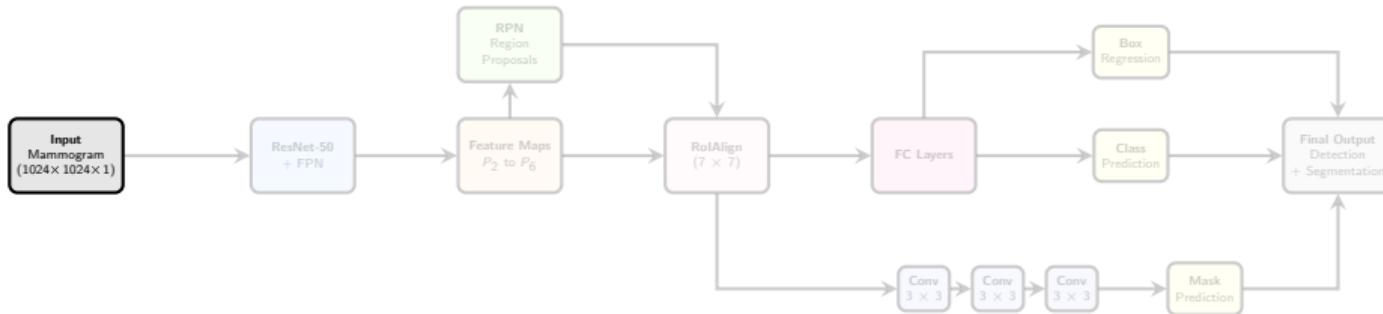
Section II

# Single-view detection

# SOTA Single-View Detection Models



# Mask R-CNN



## Step 1: Data Preparation

### Data Loading

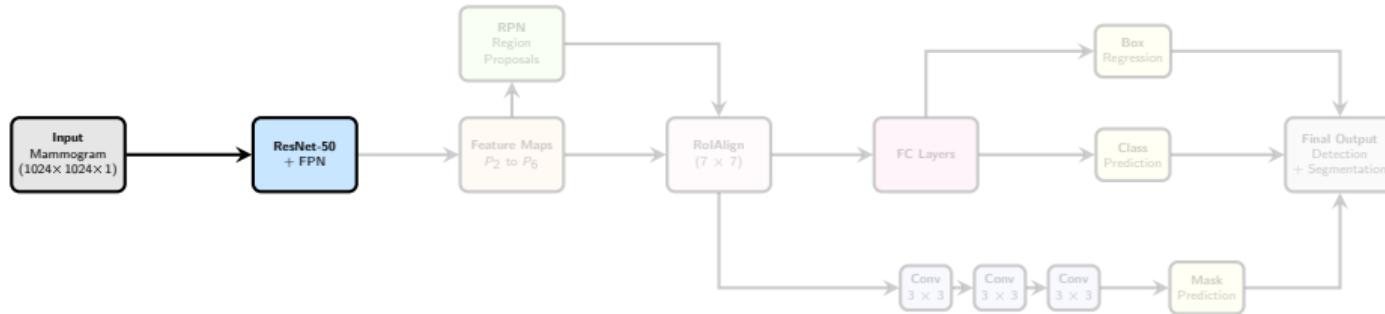
#### Mammogram Standardization:

- Let  $I_{\text{raw}} \in \mathbb{N}^{H \times W}$  coded in uint16
- Normalized image:  $I_{\text{norm}} = \frac{I_{\text{raw}}}{65535}$  then  
$$I_{\text{norm2}} = \frac{I_{\text{norm}} - m_{\text{imagenet}}}{\sigma_{\text{imagenet}}}$$
- Resized to  $1024 \times 1024$

### Medical Augmentation Pipeline

- **Elastic Distortion:** tissue deformation and compression variations
- **RandomGamma:** exposure variations between mammographs
- **GaussianBlur:** focus variations
- **GaussNoise:** sensor noise
- **RandomBrightnessContrast:** illumination differences
- **Geometric transformations:** handle different orientations

# Mask R-CNN

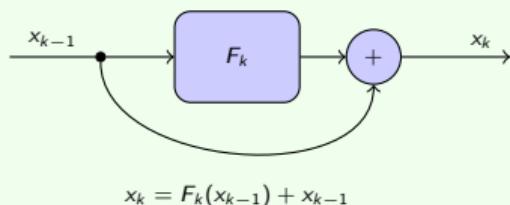


## Step 2: Backbone Feature Extraction (ResNet-50)

### ResNet-50

**Backbone:** Extracts high-level semantic features using a ResNet-50 pretrained on ImageNet.

**Residual Blocks:** Solve the vanishing gradient problem by adding skip connections.



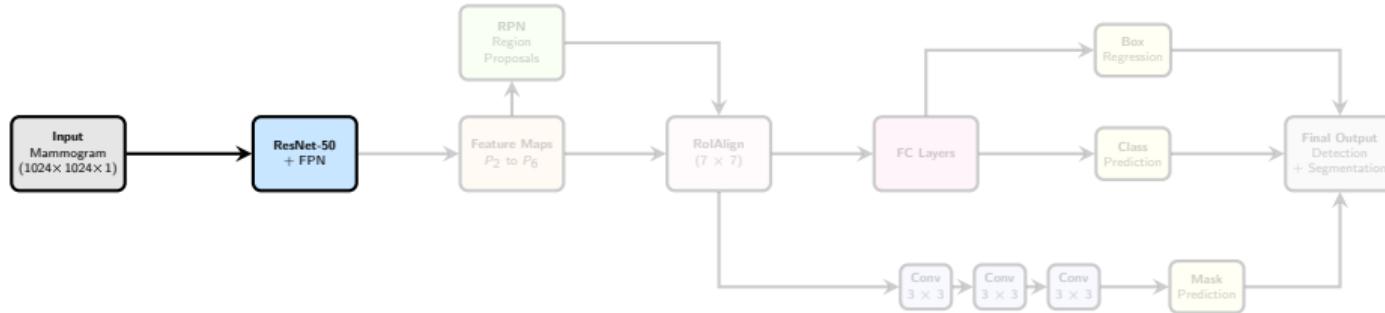
### Transform Parameters

#### Input Adaptation for Grayscale:

- ImageNet weights assume RGB input.
- Grayscale adaptation via channel-wise averaging:

$$W_{\text{gray}} = 0.299 \cdot W_R + 0.587 \cdot W_G + 0.114 \cdot W_B$$

# Mask R-CNN



## Step 3: Feature Pyramid Network (FPN)

### Feature Pyramid Network

**Bottom-up:**  $C_2$  to  $C_5$  with strides {4, 8, 16, 32}

**Top-Down Path:**  $P_i = C_i + \text{Upsample}(P_{i+1})$

**Multi-Scale Features:**

- $P_2$ : Microcalcifications (5-15px)
- $P_3$ : Small masses (15-40px)
- $P_4$ : Medium masses (40-100px)
- $P_5$ : Large masses (100-250px)

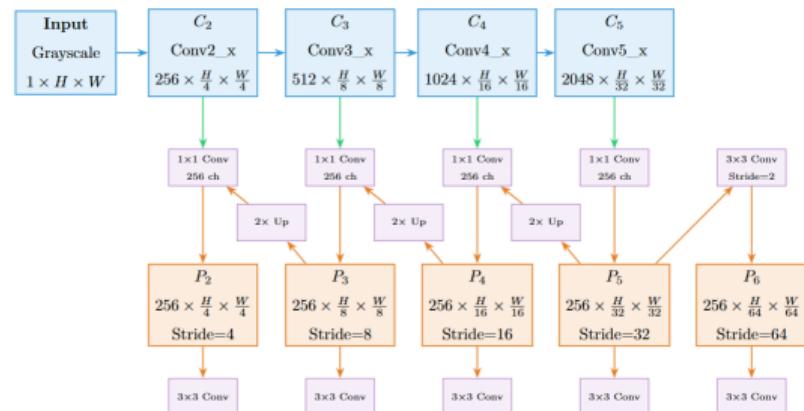
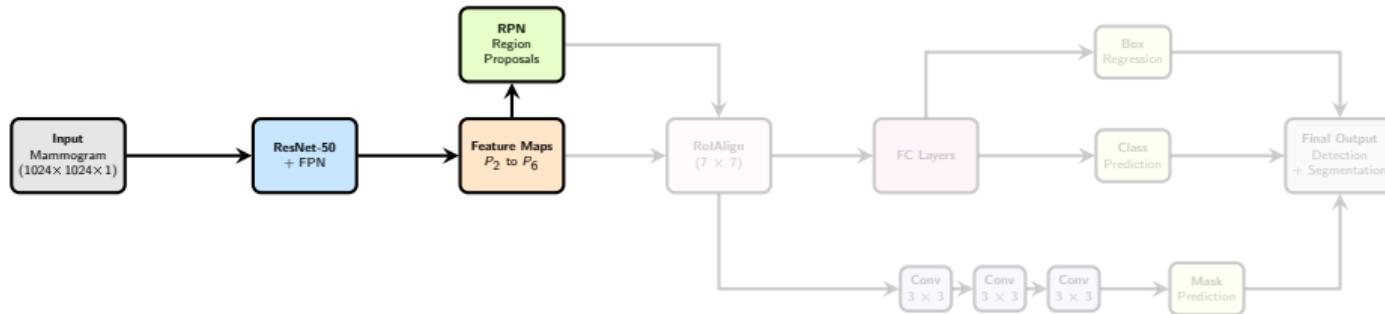


Figure: FPN architecture overview

# Mask R-CNN



## Step 4: Region Proposal Network (RPN)

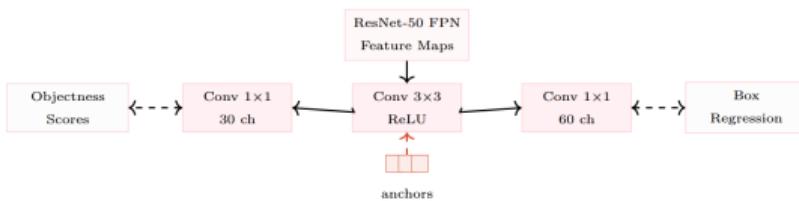


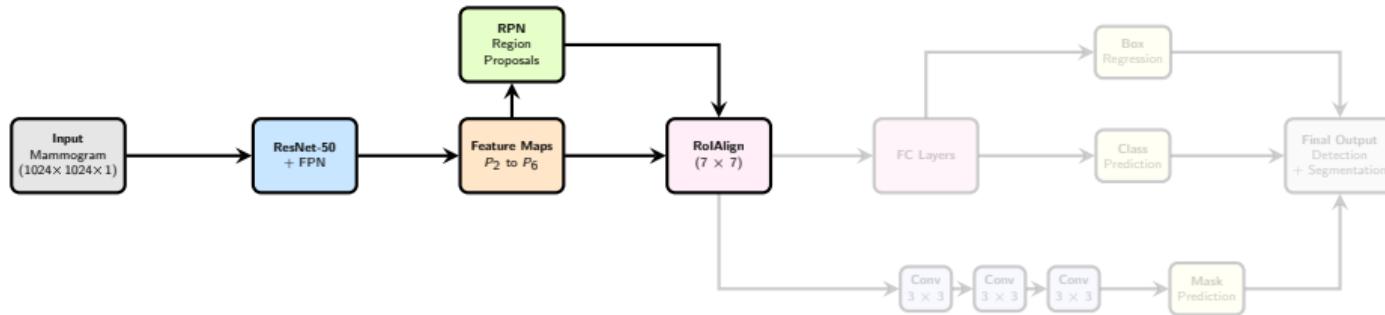
Figure: RPN architecture overview

### RPN Loss Function

$$L_{rpn} = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \frac{\lambda}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

- $L_{reg}(t, t^*) = \text{smooth}_{\ell_1}(t - t^*)$
- $L_{cls}(p_i, p_i^*) = -p_i * \log(p_i) - (1 - p_i^*) \log(1 - p_i)$

# Mask R-CNN



## Step 5: RoIAlign

### RoIAlign Implementation

#### Bilinear Interpolation:

$$f(x, y) = \sum_{i,j} f(i, j) \max(0, 1 - |x - i|) \max(0, 1 - |y - j|)$$

#### Quantization-Free Alignment:

- No coordinate snapping
- Sub-pixel accuracy maintained
- 7x7 output resolution
- Critical for mask precision

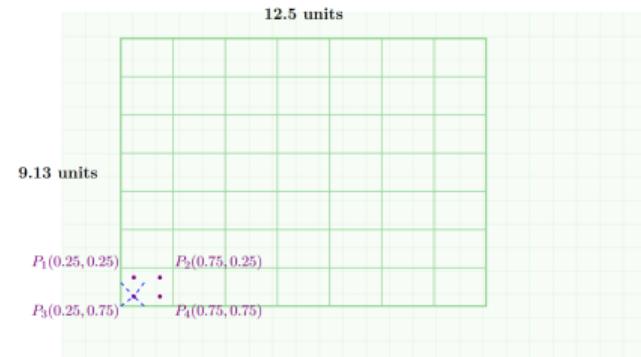
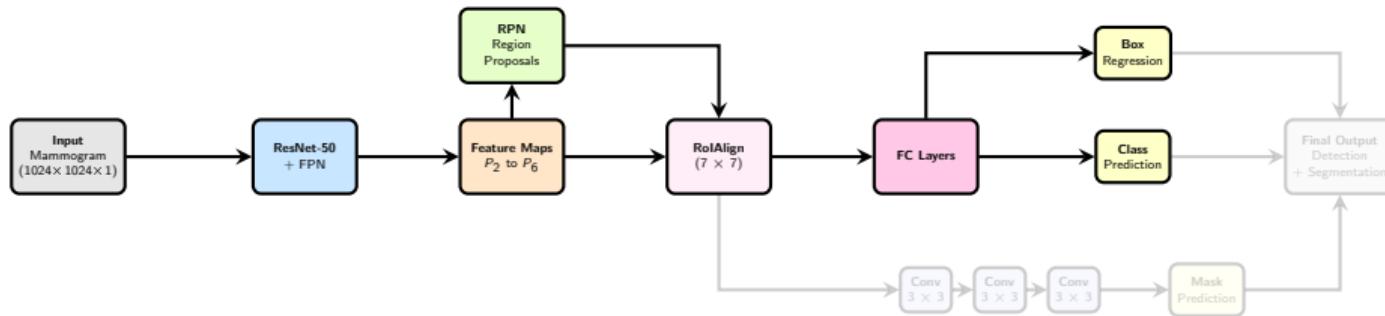


Figure: RoIAlign illustration example

# Mask R-CNN



## Step 6: Classification and Box Regression

### Classification + Box Regression

#### FC Architecture:

$$p = \text{softmax}(W_{cls} \cdot f + b_{cls})$$

$$\Delta = W_{box} \cdot f + b_{box}$$

#### Network Structure:

- FC1:  $7 \times 7 \times 256 \rightarrow 1024$
- FC2:  $1024 \rightarrow 1024$
- Class output:  $1024 \rightarrow 2$  (background + mass)
- Box output:  $1024 \rightarrow 4$  coordinates

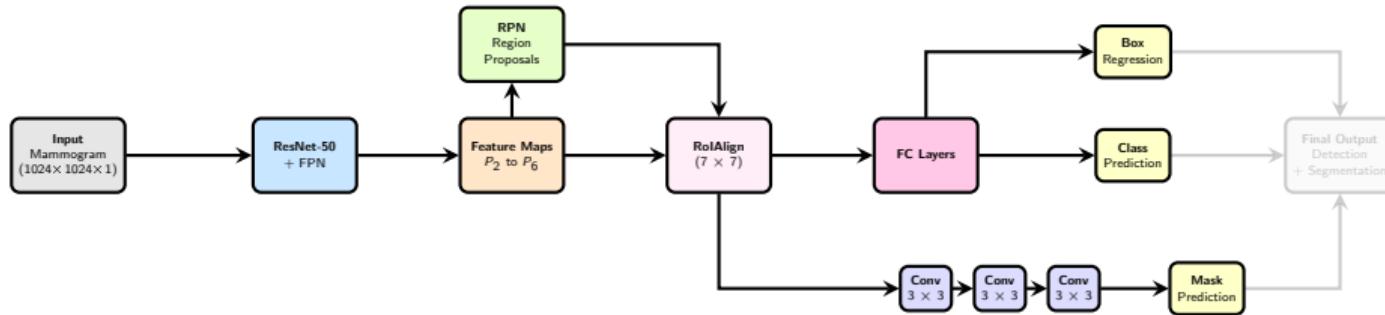
### Box Parameterization

#### Regression Targets:

- $t_x = (x - x_a) / w_a$
- $t_y = (y - y_a) / h_a$
- $t_w = \log(w / w_a)$
- $t_h = \log(h / h_a)$

**Smooth  $\ell_1$  and CE as losses**

# Mask R-CNN



## Step 7: Segmentation

### Mask Head Architecture

#### Convolutional Stack:

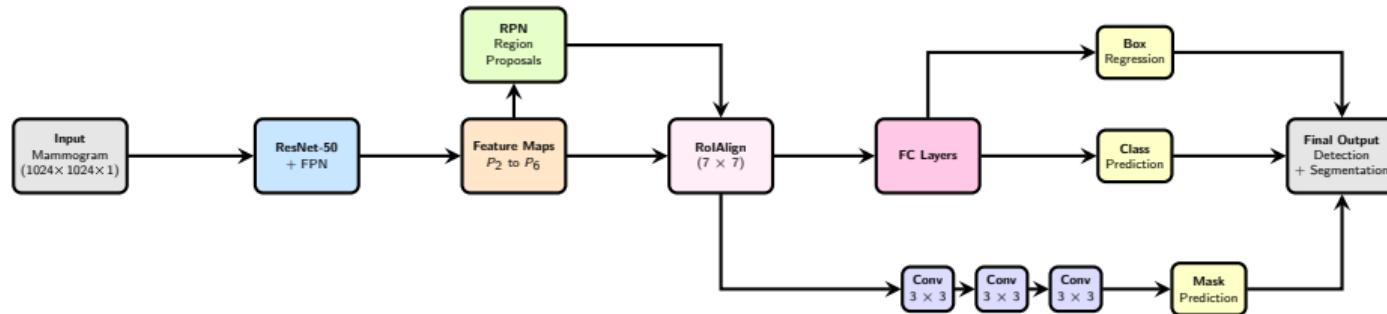
- 4x Conv3x3-256-ReLU layers
- Deconv2x2: 256 → 2 classes
- Output:  $14 \times 14 \times 2$  binary masks
- Per-class mask prediction
- Sigmoid activation for binary output

### Mask Loss

$$L_{mask} = -\frac{1}{m^2} \sum_{i,j} [y_{ij} \log(\hat{y}_{ij}) + (1 - y_{ij}) \log(1 - \hat{y}_{ij})]$$

- Computed only for positive ROIs
- Binary cross-entropy per pixel

# Mask R-CNN



## Step 8: Training on Multi-Task Loss

### Multi-Task Loss

$$L_{total} = L_{rpn} + L_{det} + L_{mask}$$

### Training Configuration:

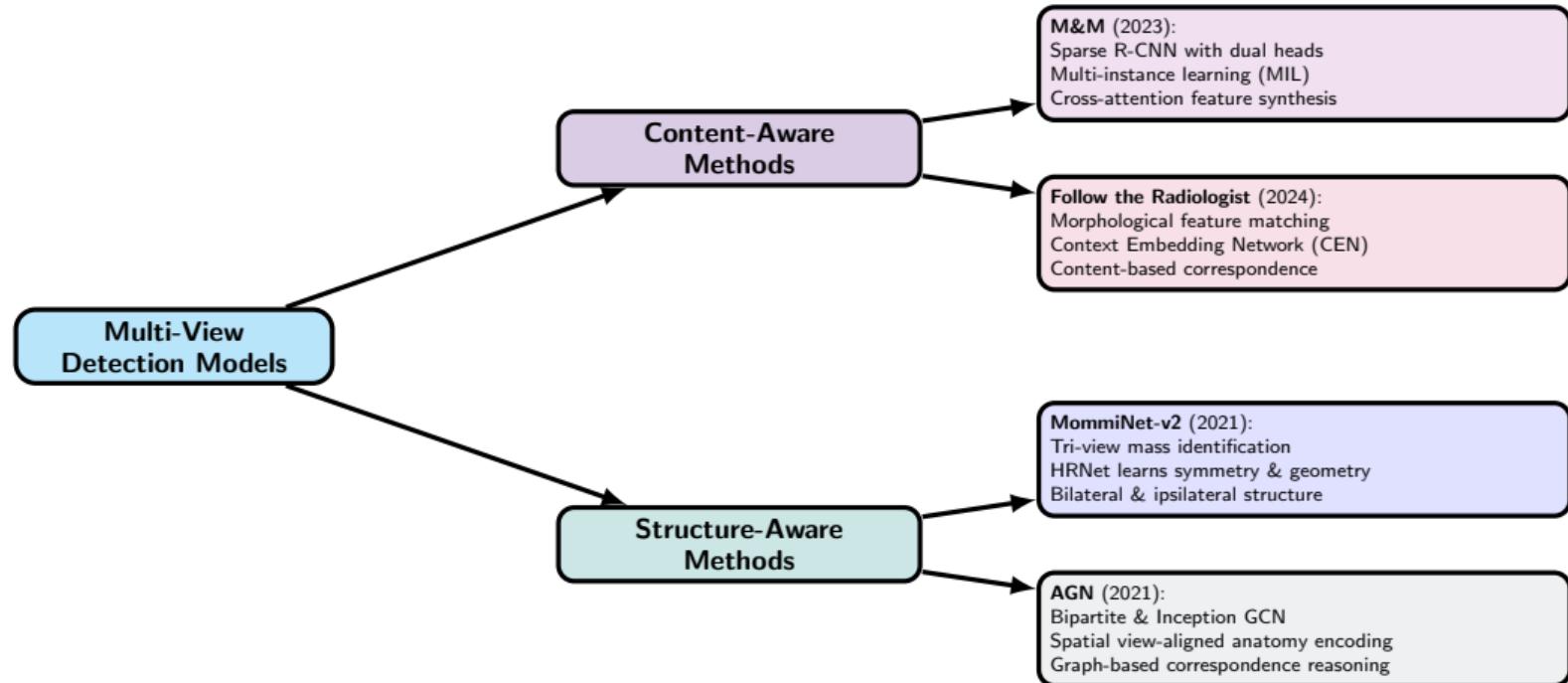
- Weight decay:  $10^{-4}$
- Momentum: 0.9
- Gradient clipping:  $\|\nabla\| \leq 1.0$
- Batch size: 2 images

### 3-stage training

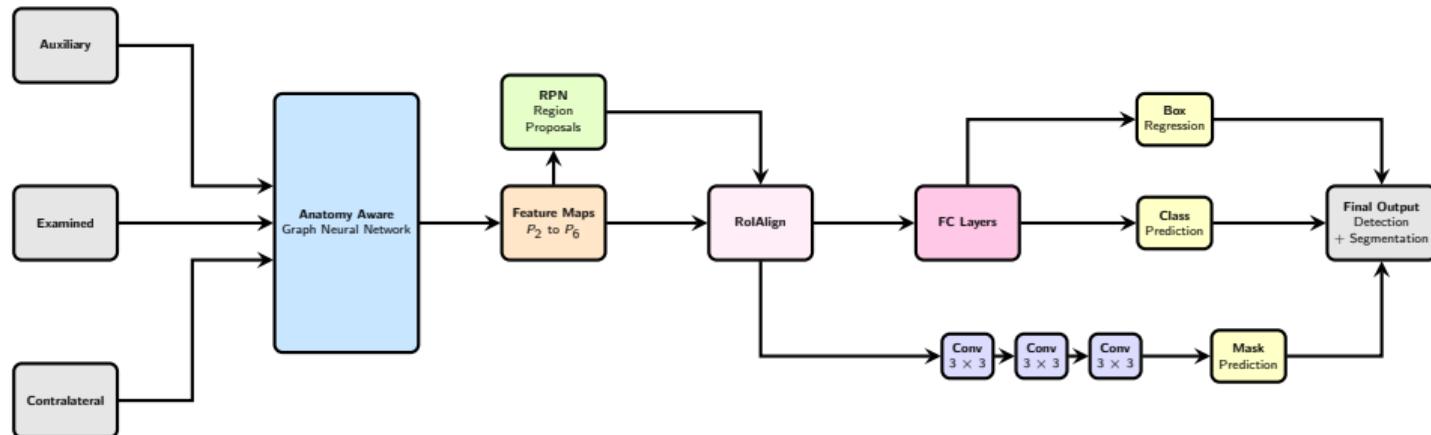
1. heads for 10 epochs (LR=0.002)
2. Initial backbone layers (LR=0.0005)
3. Full network for 5 epochs (LR=0.0001)

### Section III

# Multi-view detection



# MaskRCNN adaptation for multi-view detection



To enable multi-view reasoning, we replace the standard backbone with an **Anatomy-Aware GCN** that fuses auxiliary/contralateral features into the examined view

# Anatomy-aware Graph Network: Core Idea

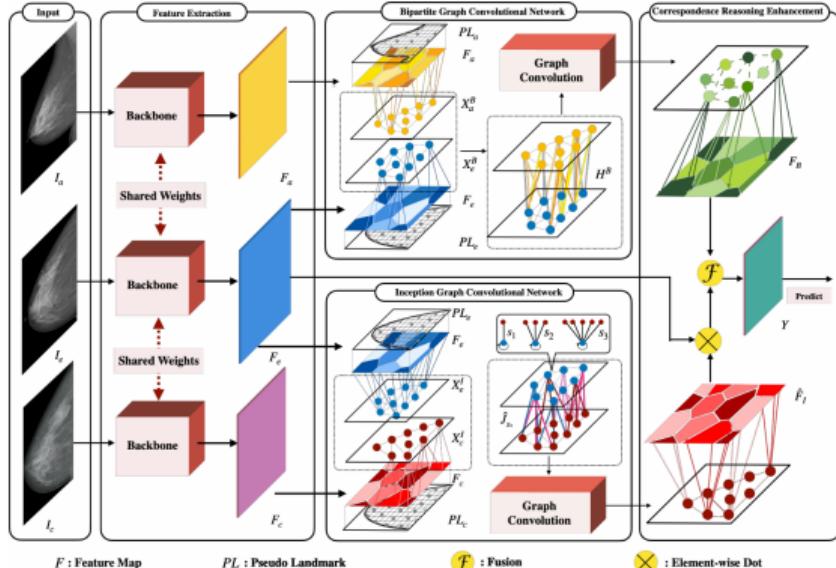


Figure: AGN architecture [Source]

## Key Idea

AGN models **multi-view anatomical reasoning** by explicitly encoding correspondences across:

- **Ipsilateral views** (CC & MLO of the same breast)
- **Bilateral views** (CC-left vs. CC-right or MLO-left vs. MLO-right)

## Key structure

The network uses two Graph Convolutional Networks (GCNs):

- **BGN:** Bipartite Graph for ipsilateral reasoning
- **IGN:** Inception Graph for bilateral symmetry analysis

## Graph construction : Pseudo landmarks

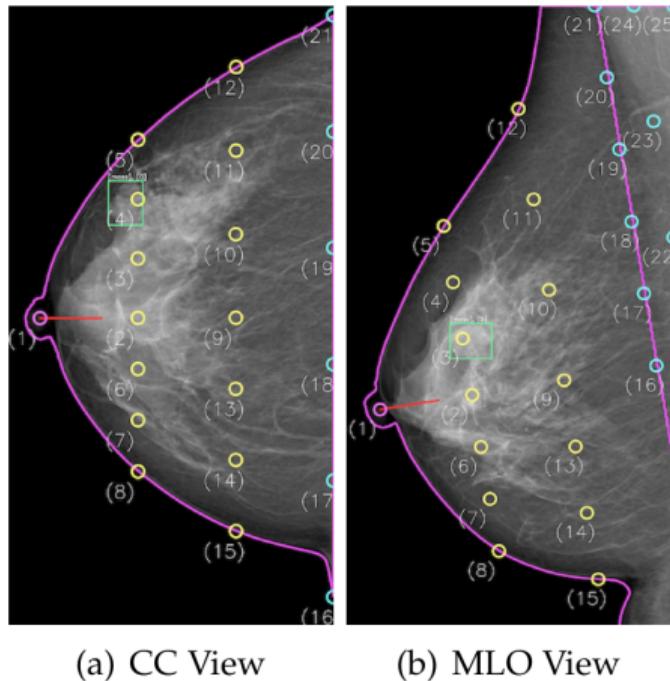


Figure: Pseudo landmarks in MLO/CC views [Source]

## Definition

Pseudo landmarks  $\mathcal{V} = \{v_i\}_{i=1}^N$  are consistent reference points on mammograms.

- Placed using anatomical priors: nipple, pectoral line and breast contour
  - Ensure consistent correspondence across patients
  - Used to define nodes in both GCNs

## Why Not Grids?

Uniform grids are sensitive to scale, shape, and orientation.  
Pseudo landmarks preserve anatomical semantics.

# Landmark Detection: Breast Contour Detection Otsu Thresholding and B-spline Smoothing

1. Breast Region Segmentation (Otsu's Method):

$$t^* = \arg \max_t \{ \omega_0(t) \omega_1(t) [\mu_0(t) - \mu_1(t)]^2 \} \quad (1)$$

2. Contour Smoothing (B-spline):

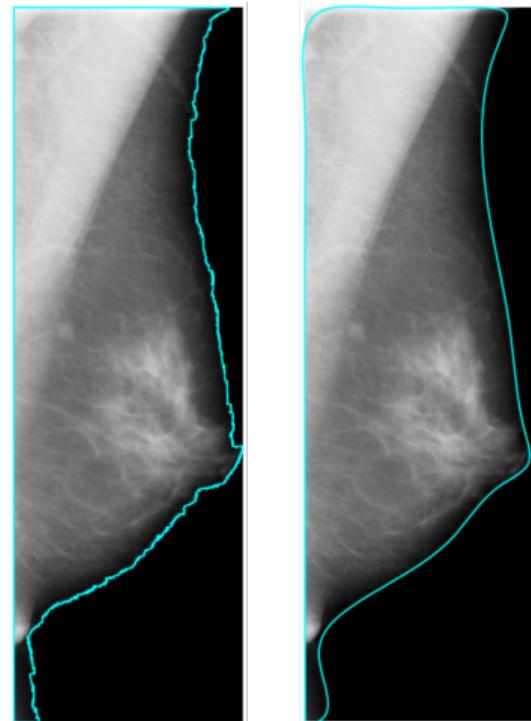
$$S(u) = \sum_{i=0}^{n-1} P_i B_i(u) \quad (2)$$

Optimization objective:

$$\min_S \left\{ \sum_{i=1}^n |C_i - S(u_i)|^2 + s \int_0^1 |S''(u)|^2 du \right\} \quad (3)$$

Adaptive smoothing parameter:

$$s = \begin{cases} 10^7 & \text{if view = MLO} \\ 100 & \text{if view = CC} \end{cases} \quad (4)$$



Contour extraction and smoothing: Raw extracted contour vs. smoothed contour with B-spline interpolation

# Landmark Detection: Nipple Detection Geometric Principles and Curvature Analysis

1. CC View Detection (Lateralmost Point):

$$p_{nipple} = \begin{cases} \arg \min_i x_i & \text{if side = Right} \\ \arg \max_i x_i & \text{if side = Left} \end{cases} \quad (5)$$

2. MLO View Detection:

Candidate Selection (Lower Lateral Quadrant):

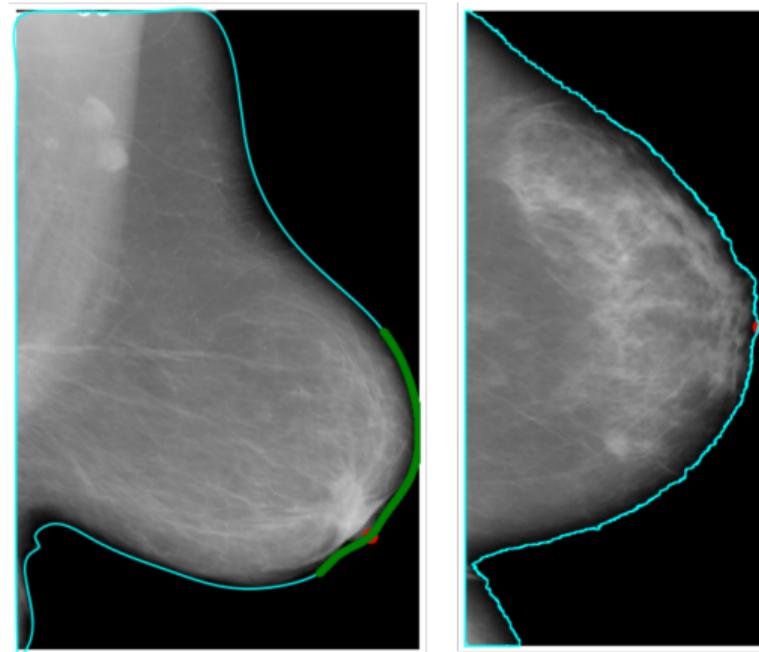
$$\begin{cases} x \geq 0.75w \text{ and } y \geq 0.5h & \text{if Left} \\ x \leq 0.25w \text{ and } y \geq 0.5h & \text{if Right} \end{cases} \quad (6)$$

Curvature Analysis:

$$\kappa(u) = \frac{x'(u)y''(u) - y'(u)x''(u)}{(x'(u)^2 + y'(u)^2)^{3/2}} \quad (7)$$

Optimal Selection:

$$\text{score}(i) = |\kappa(u_i)| \quad (8)$$



Nipple detection: (a) MLO curvature analysis with candidates (green) and final nipple (red) (b) CC lateralmost point detection

# Landmark Detection: Pectoral Muscle Detection Multi-Stage Line Detection and Scoring

## 1. CC View (Vertical Line Approximation):

$$x_{\text{pectoral}} = \begin{cases} \min_i x_i & \text{if side = Left} \\ \max_i x_i & \text{if side = Right} \end{cases} \quad (9)$$

## 2. MLO View (8-Stage Pipeline):

### ROI Definition:

$$\text{ROI} = \begin{cases} [0, 0.4w] \times [0, 0.6h] & \text{if Left} \\ [0.6w, w] \times [0, 0.6h] & \text{if Right} \end{cases} \quad (10)$$

### CLAHE Enhancement:

$$I_{\text{CLAHE}} = \text{CLAHE}(I_{\text{ROI}}, \text{clipLimit} = 3.0, \text{tileGridSize} = (8, 8)) \quad (11)$$

### Combined Thresholding:

$$T_{\text{Combined}} = \text{Otsu}(I_{\text{CLAHE}}) \wedge \text{AdaptiveThreshold}(I_{\text{CLAHE}}) \quad (12)$$

### Hough Line Detection:

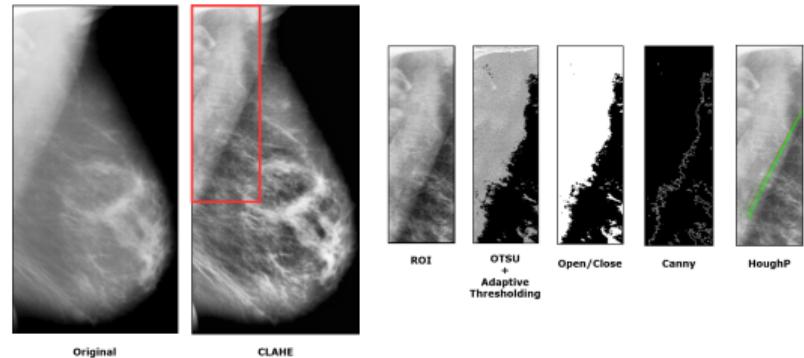
$$L = \text{HoughLinesP}(E, \rho = 1, \theta = \pi/180, \text{threshold} = 20) \quad (13)$$

### Slope Filtering:

$$\text{valid}(L_i) = \begin{cases} \text{slope}(L_i) < 0 & \text{if Left} \\ \text{slope}(L_i) > 0 & \text{if Right} \end{cases} \quad (14)$$

### Line Scoring:

$$\text{score}(L_i) = \text{length}(L_i) \cdot (w_{\text{pos}} \cdot \text{pos\_score} + w_{\text{angle}} \cdot \text{angle\_score}) \quad (15)$$



Pectoral muscle detection pipeline: (a) Original MLO (b) ROI+CLAHE (c) Thresholding (d) Morphological ops (e) Edge detection (f) Line candidates (green) and final line (red)

# Landmark Detection: Graph Construction

## Parallel Lines and k-NN Node Mapping

### 1. Parallel Line Generation:

$$\vec{v}_{pect} = \frac{\vec{p}_{pect2} - \vec{p}_{pect1}}{|\vec{p}_{pect2} - \vec{p}_{pect1}|}$$

$$\vec{p}_{line1} = \vec{p}_{nipple} + \frac{1}{3} |\vec{p}_{intersect} - \vec{p}_{nipple}| \cdot \vec{v}_{nipple-pect}$$

$$\vec{p}_{line2} = \vec{p}_{nipple} + \frac{2}{3} |\vec{p}_{intersect} - \vec{p}_{nipple}| \cdot \vec{v}_{nipple-pect}$$

### 2. Corner Line (MLO Views):

$$\vec{p}_{corner} = \begin{cases} (0, 0) & \text{if side = Left} \\ (w - 1, 0) & \text{if side = Right} \end{cases}$$

$$\vec{p}_{corner\_line} = \vec{p}_{pect\_top} + \frac{1}{2} |\vec{p}_{corner} - \vec{p}_{pect\_top}| \cdot \vec{v}_{perpendicular}$$

### 3. Node Distribution:

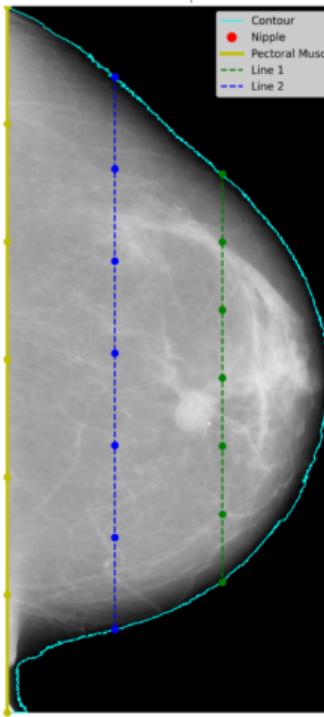
$$\vec{p}_{node\_i} = \vec{p}_{start} + \frac{i}{k-1} (\vec{p}_{end} - \vec{p}_{start}), \quad i \in \{0, 1, \dots, k-1\}$$

### 4. k-NN Node Mapping:

$$\phi_k(F, V) = (Q_f)^T F \quad Q_f = A(\Lambda_f)^{-1}$$

$$A_{ij} = \begin{cases} 1 & \text{if } j\text{th node is among } k \text{ nearest of } i\text{th pixel} \\ 0 & \text{otherwise} \end{cases}$$

(16)



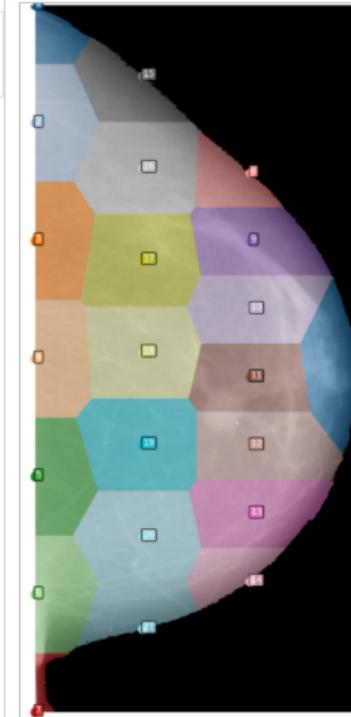
(17)

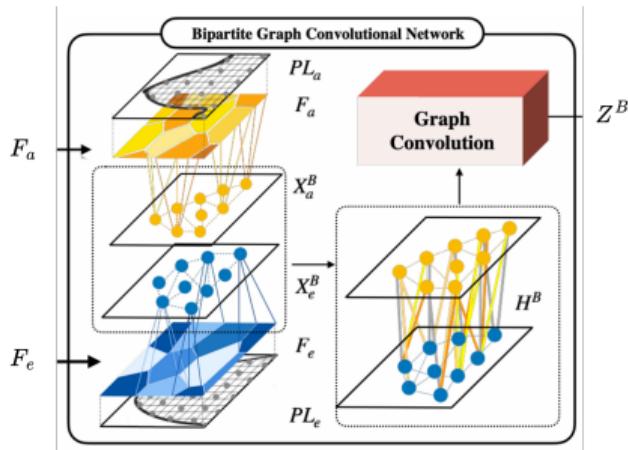
(18)

(19)

(20)

Graph construction: (a) CC view with landmarks and parallel lines (b) k-NN segmented regions mapping spatial features to graph nodes





## BGN Structure

Bipartite Graph Network models relationships between corresponding regions in CC and MLO views:

- **Nodes:** Pseudo landmarks in each view
- **Edges:** Connect nodes across views (CC to MLO)

## Feature Extraction

Node features from backbone:

$$X_e^B = \phi_k(F_e, \mathcal{V}_{l_e}), \quad X_a^B = \phi_k(F_a, \mathcal{V}_{l_a})$$

## Edge Weighting

Combines geometric and semantic information:

$$H = H_g \circ H_s$$

## Geometric Relations ( $H_g$ )

Statistical co-occurrence from training data:

- Mass instances guide correspondence
- Normalized to prevent skew:

$$H_{ij}^g = \frac{\epsilon_{ij}}{\sqrt{D_i \cdot D_j}}$$

## Semantic Relations ( $H_s$ )

Learnable appearance similarities:

- Feature fusion to model similarity:

$$H_{ij}^s = \sigma \left( \left[ \left( X_i^{CC} \right)^T, \left( X_j^{MLO} \right)^T \right] w_s \right)$$

- Adapts to individual patient characteristics

## Graph Message Passing

Information propagation across views via adjacency matrix:

$$Z^B = \sigma(H^B X^B W^B)$$

# BGN Components

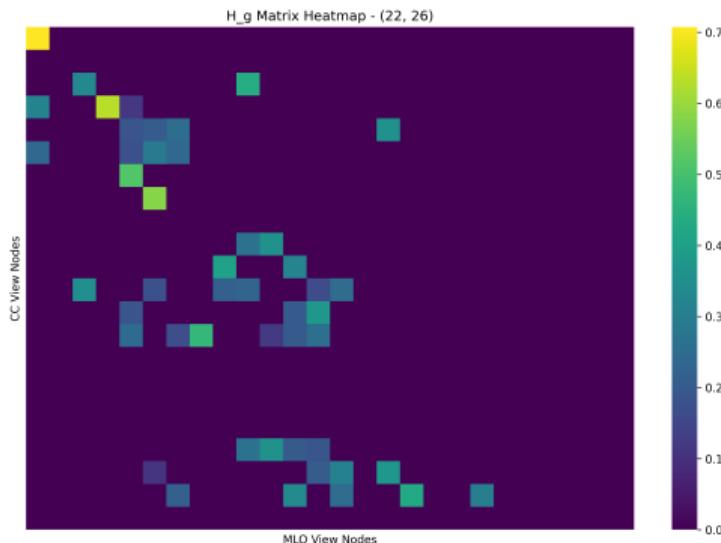
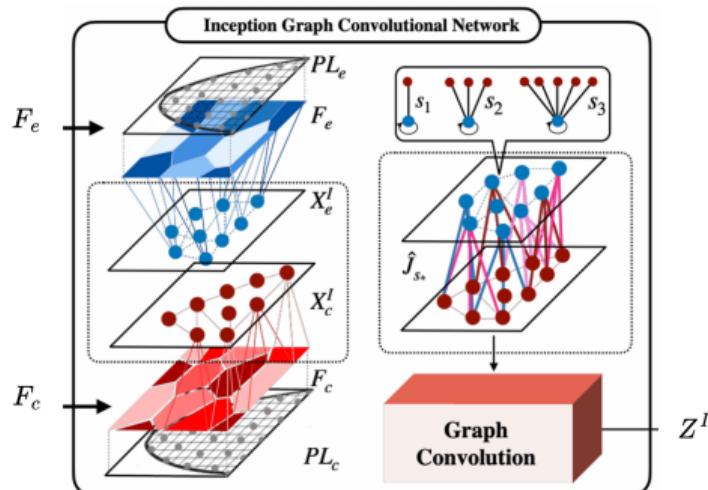


Figure: Geometric similarity matrix heatmap resulting from frequency calculation over the CBIS-DDSM training split

- We find that the nipple area  $H_{11}^g$  has the strongest correlation.
- We also find that there is no association in the pectoral muscle nodes, indicating that the presence of a mass in this location is unlikely.
- This matrix is calculated in advance which is time and memory efficient.



## Key Insight

Asymmetry between left and right breasts is a key radiological clue:

- Healthy breasts show structural symmetry
- Suspicious masses create asymmetry

## Inception Architecture

Multi-branch connections handle geometric distortions:

- $s_1, s_2, s_3$  nearest neighbor branches
- Each connects different neighbor counts
- Tolerates normal anatomical variation

## Multi-branch Adjacency ( $J_s$ )

Inception architecture with multiple neighborhood sizes:

- Each branch  $s_i$  connects to top- $s_i$  nearest neighbors
- Tolerance for geometric distortions:

$$J_{s_i}(m, n) = \begin{cases} 1 & \text{if } n \in \text{top-}s_i\text{NN}(m) \\ 0 & \text{otherwise} \end{cases}$$

## Asymmetry Detection

Output highlights regions showing bilateral asymmetry:

- Attention maps guide detection
- Robust across breast densities
- Tolerates anatomical variations

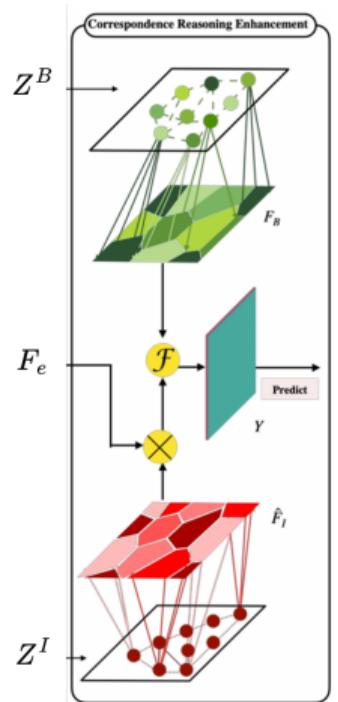
## Graph Message Passing

Multi-branch information propagation:

$$Z^I = \sigma \left( (\hat{J}_{s_1} \quad \hat{J}_{s_2} \quad \hat{J}_{s_3}) \begin{pmatrix} X^I & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & X^I & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & X^I \end{pmatrix} \begin{pmatrix} W_1^I \\ W_2^I \\ W_3^I \end{pmatrix} \right)$$

Where  $X^I = [(X_e^I)^T, (X_c^I)^T]^T$  combines examined and contralateral node features.

# Graph to Spatial Projection



## kNN Reverse Mapping $\psi_k$

Projects node features back to image space:

$$F^B = \psi_k(Z^B, \mathcal{V}_e), \quad F^I = \psi_k(Z^I, \mathcal{V}_e)$$

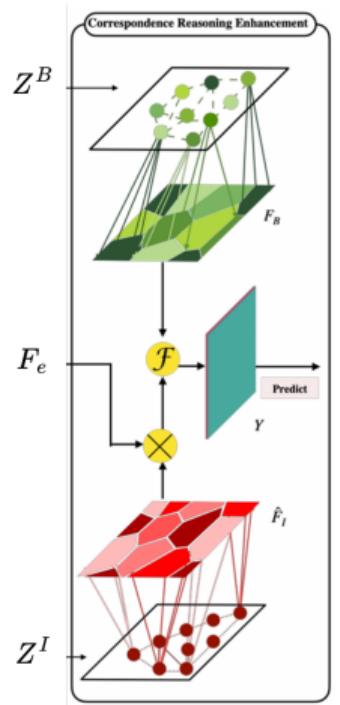
For each pixel, weighted average of k-nearest nodes.

## Attention Application

IGN produces spatial attention map for examined view:

$$\hat{F}^I = \sigma(F^I w_I)$$

Highlights regions showing asymmetry with contralateral breast.



## Final Feature Enhancement

Combined multi-view reasoning:

$$Y = [\hat{F}_I \circ F_e, F_B] W_f^\top$$

Where:

- $\hat{F}_I \circ F_e$ : Attention-weighted examined features
- $F_B$ : Ipsilateral correspondences
- $W_f$ : Fusion layer parameters

## Section IV

# Results & Comparison

# Dataset Overview: CBIS-DDSM

## Statistical Composition and Distribution

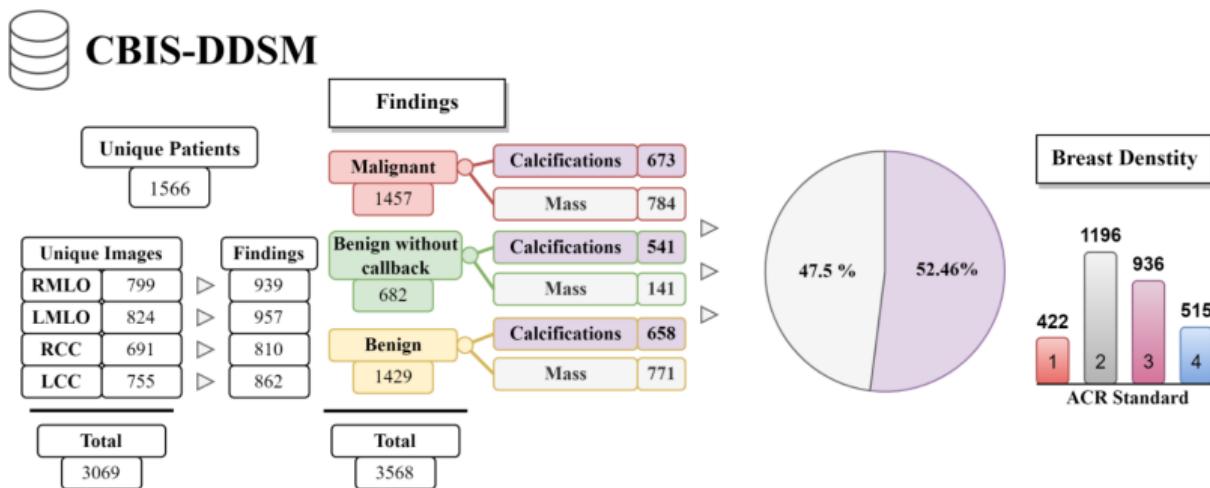


Figure: The CBIS-DDSM database statistics [Source]

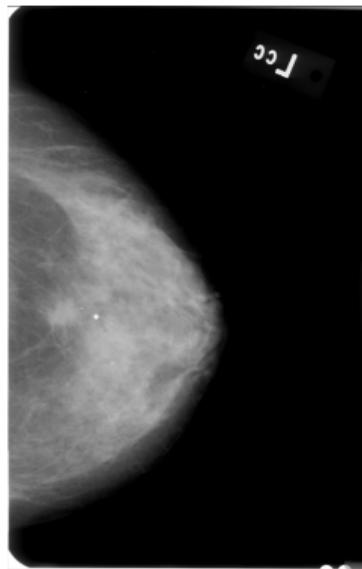
**Key Statistics:** 1,566 unique patients • 3,069 mammographic images • 3,568 annotated findings

**Finding Distribution:** 1,457 malignant cases, 2,111 benign cases across four mammographic views (RMLO, LMLO, RCC, LCC) with varying breast density classifications (ACR 1-4).

# Dataset Overview: File Organization

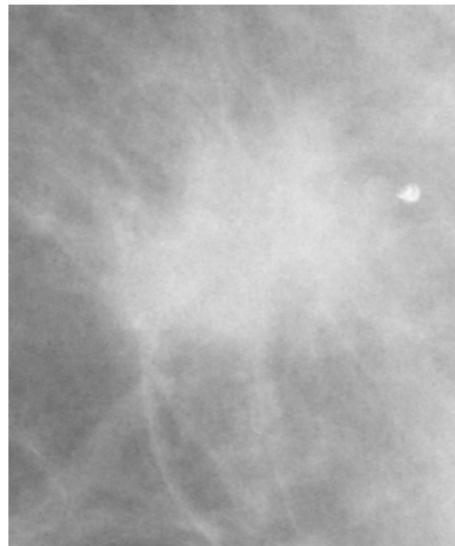
## Understanding the Three-Component System

**Mammogram**



Original breast X-ray image

**ROI**



Cropped area

**Mask**



Binary segmentation mask

## Loss analysis: MaskRCNN End-to-end vs. staged training

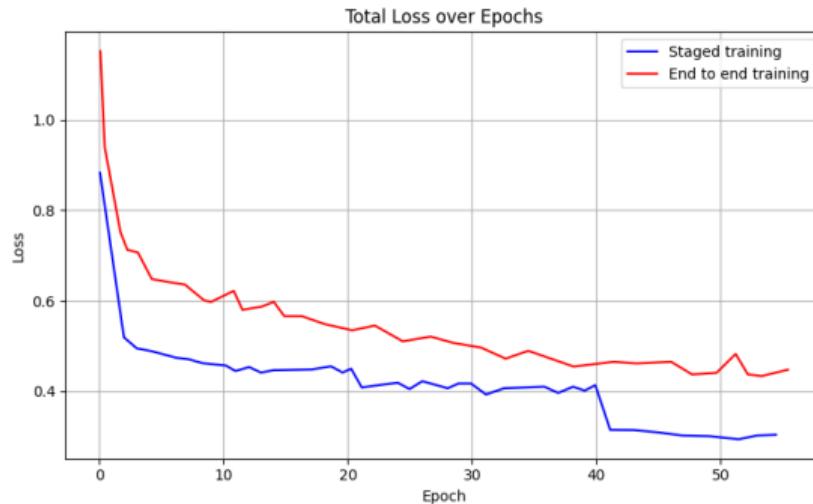


Figure: Total loss comparison between 3-stage training and end-to-end training

- **Staged Training Advantage:** Drops at each transition from one stage to another, rather than early plateau
- **Better Convergence:** Staged training achieves lower final loss compared to end-to-end approach
- **Training Stability:** Noise due to small batch size (2) for memory constraints accommodation
- **Progressive Learning:** Each stage builds upon previous knowledge systematically

# Loss analysis: MaskRCNN

## Training vs. Validation

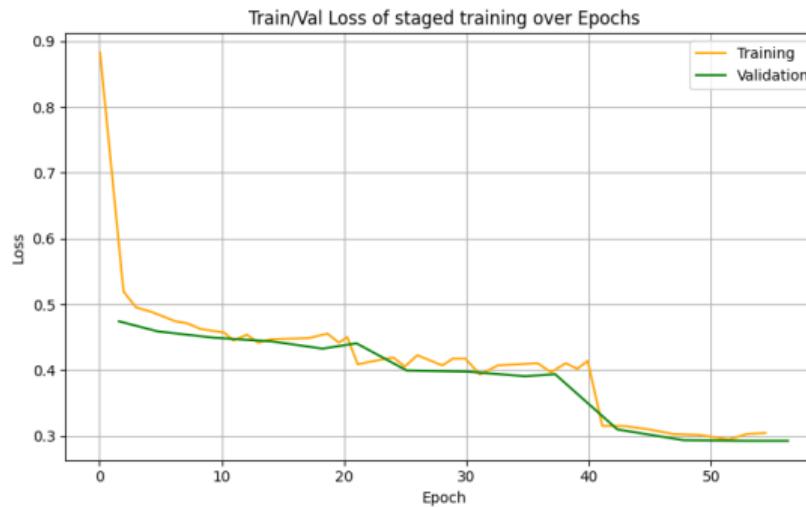


Figure: Comparison between training and validation losses in 3-stage training

- **Proper Fitting:** Validation loss follows training loss pattern
- **Smoother Validation:** Computed every two epochs to optimize computational resources
- **Convergence Success:** Both losses stabilize at acceptable levels

# AGRCNN analysis

## Initial training problem

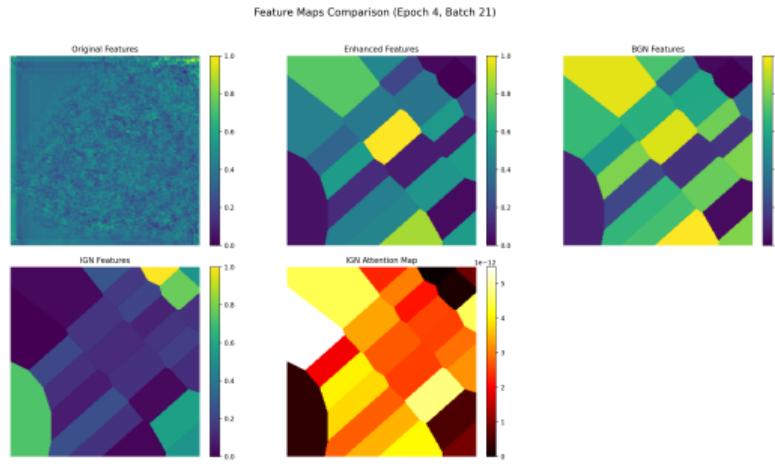


Figure: AGN Features with staged training before model adjustments

- **Destructive Attention:** IGN attention values approaching zero (scale:  $10^{-12}$ )
- **Feature Elimination:** Original multiplicative approach:  $F_{enhanced} = \sigma(F_I w_I) \odot F_e$
- **Training Instability:** Randomly initialized weights disrupted pre-trained MaskRCNN features
- **Performance Degradation:** Unable to recover initial MaskRCNN performance levels

# AGRCNN analysis

## AGN Feature Enhancement

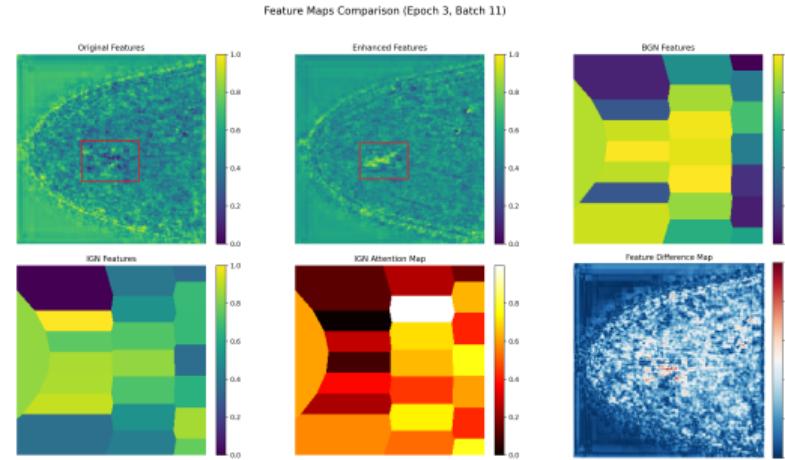


Figure: AGN Features with staged training after model adjustments

- **Background Suppression:** Enhanced features show reduced activation in irrelevant regions (breast contour, background)
- **Mass Enhancement:** Target lesion regions exhibit stronger, more focused activation
- **Residual Attention:**  $F_{enhanced} = F_e \odot (2\hat{F}_I + 0.2)$  enables both suppression and enhancement
- **Feature Preservation:** 20% residual connection maintains base features even with minimal attention

# AGRCNN analysis

## AGRCNN loss

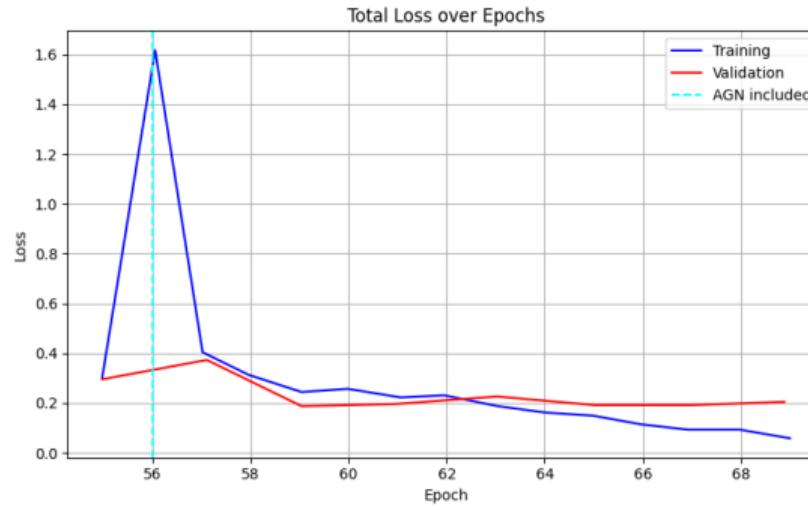


Figure: Training and Validation losses after AGN integration

- **Integration Point:** Training begins at epoch 56 (post-MaskRCNN training)
- **Initial Spike:** Loss peak due to arbitrary feature representation impact at start
- **Rapid Improvement:** Significant loss decrease achieving better results than MaskRCNN alone
- **Early Stopping:** Implemented at epoch 62 due to overfitting (87 training, 24 validation samples)

# AGRCNN analysis

## Ablation studies

Table: Component-wise Performance Analysis on CBIS-DDSM (%).

Method	R@0.5	R@1.0	R@2.0	R@3.0	R@4.0
MaskRCNN (Baseline)	68.9	79.8	86.3	90.2	91.3
+ BGN only	72.1	81.5	87.8	90.8	91.7
+ IGN only	71.3	82.2	88.1	90.5	91.9
+ AGN (Original fusion)	54.2	63.1	68.9	71.1	72.0
<b>+ AGN (Our modifications)</b>	<b>78.4</b>	<b>85.5</b>	<b>90.1</b>	<b>91.6</b>	<b>92.5</b>

Table: Pseudo-Landmark Density Analysis on CBIS-DDSM (%).

Configuration	R@0.5	R@1.0	R@2.0	Notes
PL(13, 17)	76.8	84.1	89.3	Sparse configuration
<b>PL(22, 26)</b>	<b>78.4</b>	<b>85.5</b>	<b>90.1</b>	<b>Optimal density</b>
PL(100, 105)	77.2	84.8	89.7	Over-parameterized

Table: Graph Node Mapping Parameter Analysis on CBIS-DDSM (%).

Mapping Strategy	R@0.5	R@1.0	R@2.0	Notes
kNN, k=1 (Voronoi)	75.2	83.8	88.9	Nearest neighbor only
<b>kNN, k=3</b>	<b>78.4</b>	<b>85.5</b>	<b>90.1</b>	<b>Optimal context</b>
kNN, k=5	77.8	84.9	89.7	Over-smoothed features

## Evaluation metrics

### Recall@FPI

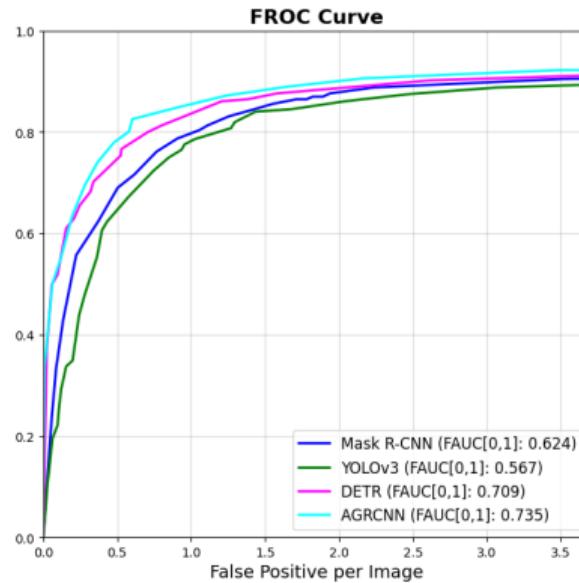


Figure: Comparative FROC analysis between MaskRCNN, YOLO, DETR and AGRCNN

- **Superior Performance:** AGRCNN clearly outperforms all single-view models
- **High Sensitivity:** Achieves high recall without generating additional false positives
- **Model Convergence:** All models converge at high FPI due to increased detection boxes

# Evaluation metrics

## Metrics comparison

Model	0.5 FPI	1.0 FPI	2.0 FPI	3.0 FPI	4.0 FPI	Dataset	Images
<i>Reference Paper (ALR) Results</i>							
ALR MaskRCNN+FPN	83.1%	88.0%	91.4%	93.4%	94.2%	DDSM	2,620
ALR AG-RCNN	87.6%	90.6%	93.4%	94.7%	95.2%	DDSM	2,620
<i>ALR Improvement</i>	+4.5%	+2.6%	+2.0%	+1.3%	+1.0%		
<i>Our Implementation Results</i>							
Our MaskRCNN+FPN	68.9%	79.8%	86.3%	90.2%	91.3%	CBIS-DDSM	1,560
Our AGRCNN	78.4%	85.5%	90.1%	91.6%	92.5%	CBIS-DDSM	1,560
<i>Our Improvement</i>	+9.5%	+5.7%	+3.8%	+1.4%	+1.2%		
<b>Difference</b>	<b>+5.0%</b>	<b>+3.1%</b>	<b>+1.8%</b>	<b>+0.1%</b>	<b>+0.2%</b>		(-40%)

Table: AGRCNN Performance Enhancement Comparison with Dataset Information

- **Consistent Enhancement:** AGN provides substantial recall improvements across all FPI levels
- **Existing bias:** MaskRCNN of ALR had less room for improvement because it has already been trained on a large dataset

Section V

# Conclusion & Perspectives

## Key Takeaways

- **Clinically Meaningful Improvements:** Multi-view detection models outperform drastically the single-view models at performance. 9.5% recall gain  $\implies \sim 47$  fewer missed cancers per 1000 screens
- **Data Challenge:** they are constrained by the requirement of big datasets with multiple mammograms per patient

## Future Directions

- Training AGRCNN on larger datasets and compare it with other SOTA multi-view methods
- Develop a malignancy classification model (ResNet+CBAM, InceptionV3, etc.)

## Opening

- Multi-view models are inherently limited as they try to predict a 3D mass from 2D projections
- This motivates the reconstruction of dense representations like DBT or MRI using neural representations and perform detection in 3D

# Thank you for your attention

## Questions?

**Student:** Imade Bouftini  
**Supervision:** Youssef ALJ

**AI Movement**  
Mohammed VI Polytechnic University  
October 20, 2025

### Appendix

- DETR architecture overview
- DDSM cleaning
- Anchors optimization in MaskRCNN
- GPU training details

# Detection Transformer (DETR)

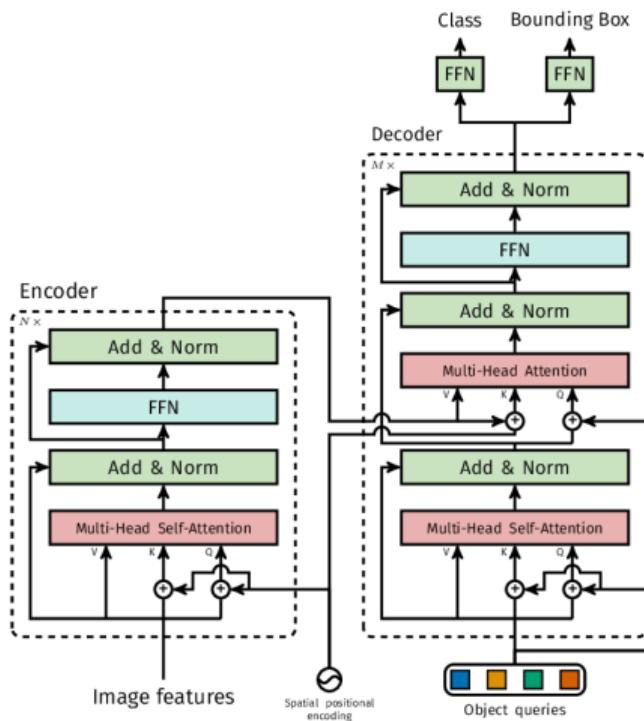


Figure: DETR architecture overview

## Step 1: Feature Extraction & Positional Encoding

### CNN Backbone

#### ResNet-50 Feature Extraction:

- Input:  $\mathbf{x}_{img} \in \mathbb{R}^{H \times W \times 1}$
- CNN features:  $\mathbf{f} \in \mathbb{R}^{H/32 \times W/32 \times C}$
- Lower resolution but rich semantics
- $C = 2048$  for ResNet-50

### Positional Encoding

#### Spatial Position Information:

$$\mathbf{f}_{final} = \mathbf{f} + \mathbf{pos}$$

- Sine/cosine positional encoding
- Essential for spatial reasoning
- $\mathbf{pos} \in \mathbb{R}^{H/32 \times W/32 \times C}$
- Enables transformer to understand spatial layout

# Detection Transformer (DETR)

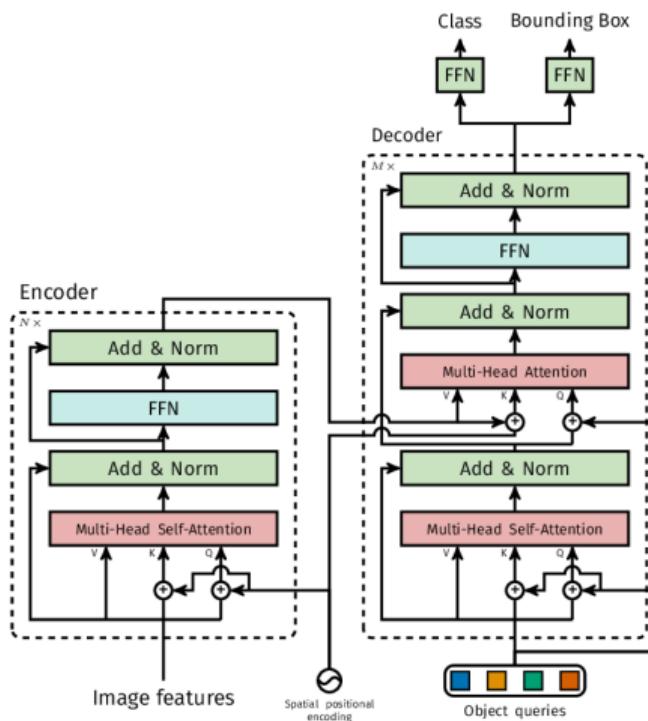


Figure: DETR architecture overview

## Step 2: Transformer Encoder

### Self-Attention Mechanism

#### Multi-Head Self-Attention:

$$\text{MSA}(\mathbf{X}) = \text{Concat}(\text{head}_1, \dots, \text{head}_h)\mathbf{W}^O$$

$$\text{head}_i = \text{Attention}(\mathbf{X}\mathbf{W}_i^Q, \mathbf{X}\mathbf{W}_i^K, \mathbf{X}\mathbf{W}_i^V)$$

Where:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax} \left( \frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}} \right) \mathbf{V}$$

### Encoder Layer Structure

$$\mathbf{z}_I = \text{MSA}(\text{LN}(\mathbf{z}_{I-1})) + \mathbf{z}_{I-1}$$

$$\mathbf{z}_I = \text{FFN}(\text{LN}(\mathbf{z}_I)) + \mathbf{z}_I$$

- Global receptive field from first layer
- $N \times$  encoder layers process image features

# Detection Transformer (DETR)

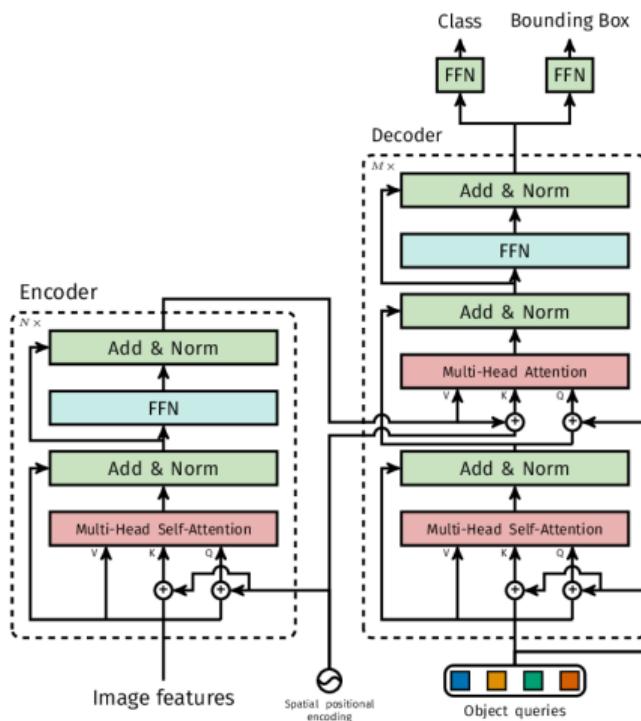


Figure: DETR architecture overview

## Step 3: Transformer Decoder & Object Queries

### Object Queries

#### Learnable Detection Slots:

- $N = 100$  learnable embeddings
- $\mathbf{q}_{obj} \in \mathbb{R}^{N \times d}$  (where  $d = 256$ )
- Each query focuses on different objects
- Learned during training to specialize

#### Query Initialization:

$$\mathbf{q}_{obj} \sim \mathcal{N}(0, \sigma^2)$$

### Decoder Layer

#### Self-Attention + Cross-Attention:

$$\mathbf{q}_I = \text{SelfAttn}(\text{LN}(\mathbf{q}_{I-1})) + \mathbf{q}_{I-1}$$

$$\mathbf{q}_I = \text{CrossAttn}(\text{LN}(\mathbf{q}_I), \mathbf{z}_{enc}) + \mathbf{q}_I$$

$$\mathbf{q}_I = \text{FFN}(\text{LN}(\mathbf{q}_I)) + \mathbf{q}_I$$

# Detection Transformer (DETR)

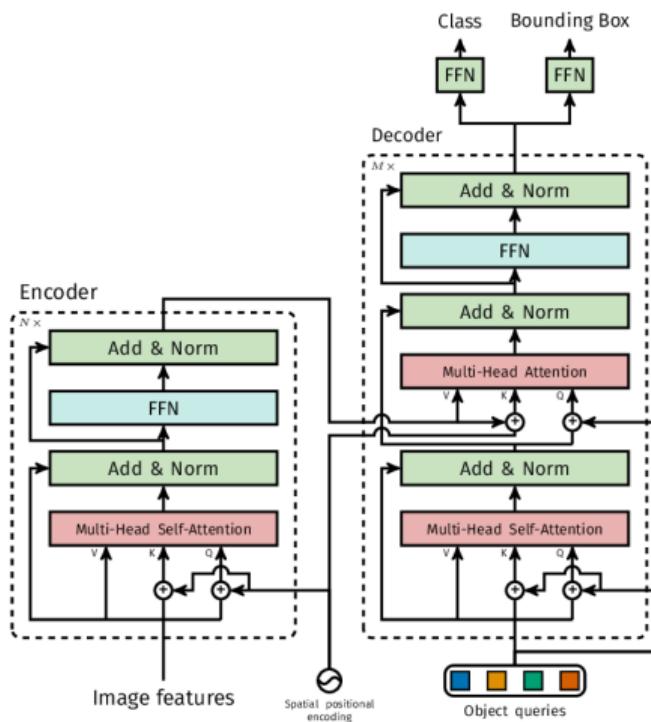


Figure: DETR architecture overview

## Step 4: Prediction & Hungarian Matching

### Prediction Heads

#### Classification Head:

$$p_i = \text{softmax}(\text{FFN}_{cls}(\mathbf{q}_i))$$

#### Box Regression Head:

$$\mathbf{b}_i = \sigma(\text{FFN}_{box}(\mathbf{q}_i))$$

- Each query produces one prediction

### Hungarian Algorithm

#### Bipartite Matching:

$$\hat{\sigma} = \arg \min_{\sigma \in \mathfrak{S}_N} \sum_i^N \mathcal{L}_{match}(y_i, \hat{y}_{\sigma(i)})$$

#### Set-based Loss:

$$\mathcal{L} = \sum_{i=1}^N [-\log \hat{p}_{\hat{\sigma}(i)}(c_i) + \{c_i \neq \emptyset\} \mathcal{L}_{box}(b_i, \hat{b}_{\hat{\sigma}(i)})]$$

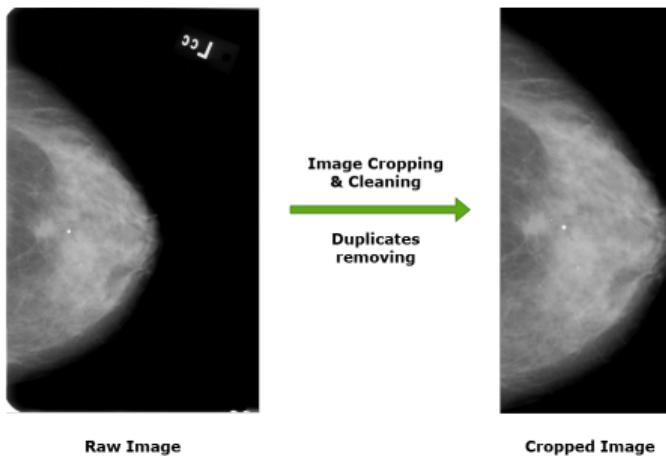
# Data cleaning: Quality Assurance

## Systematic Dataset Cleaning and Correction

The CBIS-DDSM dataset presented several critical inconsistencies requiring systematic correction

### Unnecessary Image Regions

- Mammograms contained irrelevant background and metadata areas
- *Solution:* Implemented cropping algorithm (provided by Mr. Yassine Ameskine) to isolate regions of interest
- *Impact:* Focused processing on clinically relevant breast tissue only

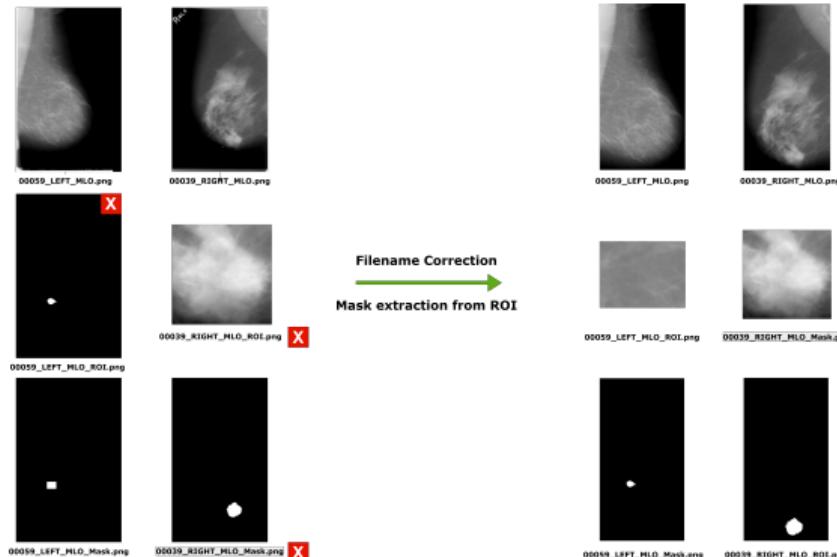


# Data cleaning: File Recovery

## Addressing Corruption and Annotation Errors

### File Corruption and Mismatched Annotations

- Swapped filenames between masks and ROI files
- Missing or deleted mask files
- *Solution:* Developed directory correction algorithm to restore proper file associations and recover masks from ROI



# More about MaskRCNN

## Threshold Optimization optimization and Trade-offs

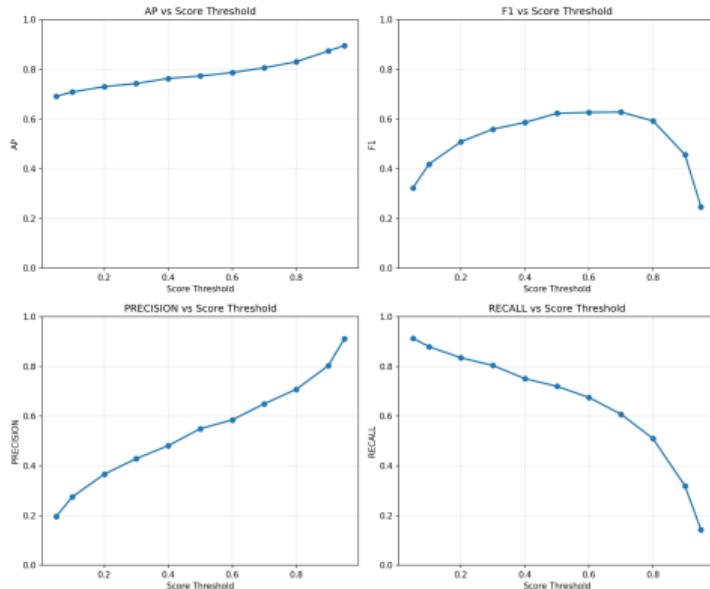


Figure: Performance metrics vs. confidence threshold

- **Optimal Threshold:** 0.7 (F1-maximized)
- **F1-Score:** 0.62
- **Precision:** 0.65
- **Recall:** 0.6
- **High-confidence predictions:** 23.2% above 0.5

### Clinical Relevance

Threshold can be adjusted based on screening vs. diagnostic priorities

# More about MaskRCNN

## Anchor Configuration Validation

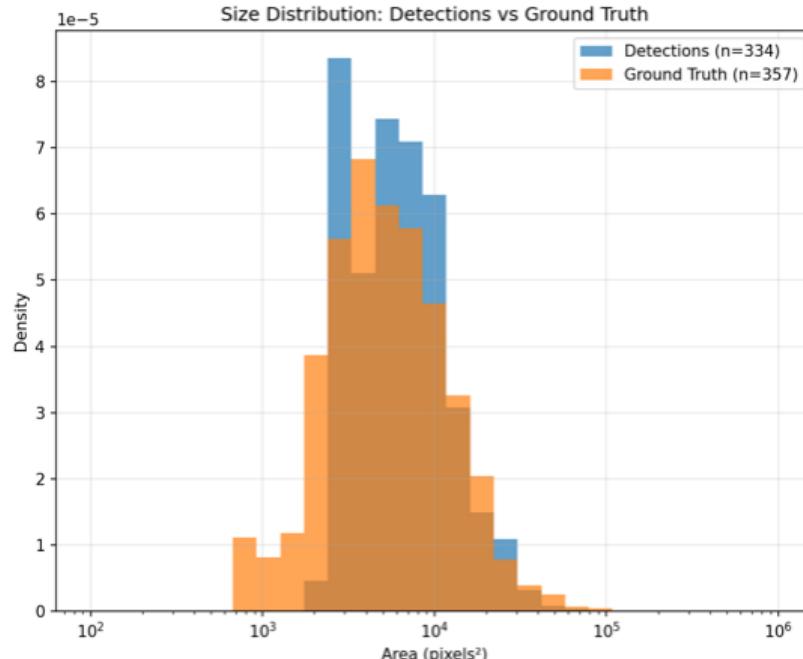


Figure: Predicted vs. ground truth size distributions

- **Detected masses:** 334
- **Ground truth:** 357
- **Size range:** 10<sup>3</sup> – 10<sup>5</sup> pixels<sup>2</sup>
- Close distribution alignment

### Validation

No bias toward large/small masses  
Successful transfer learning