1. METHODOLOGY

A. Data Processing Pipeline

Our TTI (Tool-Tissue Interaction) detection system employs a multi-stage pipeline that processes laparoscopic video data to identify and classify surgical interactions. Each video had the beginning and end of interaction along with their tools annotated. The methodology encompasses several key components:

a. YOLOv11 Data Processing

The YOLOv11 model requires specialized data preprocessing for tool and tissue segmentation tasks. Our dataset comprises laparoscopic surgical videos with pixel-level annotations for both surgical tools and tissue interactions.

Input Specifications

- Video Format: MP4 files containing laparoscopic surgical procedures
- Frame Extraction: Videos processed at 30 FPS with frame-by-frame extraction
- Frame Dimensions: Variable resolution (typically 1920×1080 or 1280×720)
- Color Space: BGR (OpenCV default) converted to RGB for processing

Preprocessing Pipeline

- Frame Normalization: Pixel values scaled to [0, 1] range
- Resizing: All frames resized to 640×640 pixels (YOLO standard input size)
- Data Augmentation: Applied during training including:
    - Oversampling to make the dataset balanced such that all types of tool and interaction classes are equally represented.
    - Random horizontal flipping (p=0.5)
    - Random rotation (±15 degrees)
    - Color jittering (brightness, contrast, saturation)
    - Random scaling (0.8-1.2x)

Annotation Format

- Segmentation Masks: Binary masks for each tool and tissue instance
- Class Labels: 12 tool classes (0-11) and 9 tissue classes (12-20)
- Bounding Boxes for Tool-Tissue Interaction Areas: Normalized coordinates [x_center, y_center, width, height]
- File Format: YOLO segmentation format with polygon coordinates

Output Specifications

The YOLO model produces structured outputs for each frame:

- Detection Results: Bounding boxes with confidence scores and class predictions
- Segmentation Masks: Binary masks for precise object boundaries
- Prototype Masks: 32 prototype masks at 160×160 resolution
- Mask Coefficients: Coefficients for combining prototypes into instance masks

b. TTI Classifier Data Processing

The TTI classifier processes Region of Interest (ROI) data extracted from tool and interaction masks obtained from the YOLOv11 model to classify whether an interaction constitutes a TTI.

ROI Extraction Process

- Tool-Tissue Pairing: Algorithm identifies potential interactions between detected tools and tissues
- Spatial Filtering: Pairs filtered based on proximity thresholds and mask overlap
- Intersection Detection: intersection_mask = tool_mask & tissue_mask
- Bounding Box Calculation: Exact bounding box of interaction area with minimum size enforcement (64×64 pixels)

ROI Preprocessing

- Dimensions: Variable based on interaction area (minimum 64×64 pixels)
- Channel Structure: 5-channel data combining:
- Channels 0-2: RGB values [0, 255]
- Channel 3: Depth values [0, 1] from depth estimation
- Channel 4: Interaction mask [0, 255]

- Resizing: ROI resized to 224×224 pixels for model input
- Normalization: Pixel values scaled to [0, 1] range
- Tensor Conversion: Shape (batch_size, 5, 224, 224) for batch processing

Depth Channel Integration

- Model: Depth-Anything-V2-Small for monocular depth estimation
- Input: RGB frame converted to PIL Image format
- Output: Single-channel depth map with same dimensions as input
- Integration: Depth channel concatenated with RGB and mask channels

B. YOLOv11 Model Architecture for Tool and Interaction Segmentation

We employ YOLOv11-seg, a state-of-the-art segmentation model specifically designed for instance segmentation tasks. The model architecture consists of:

a. Backbone Network:

- Feature Extractor: CSPDarknet backbone with cross-stage partial connections
- Input Dimensions: 640×640×3 RGB images
- Feature Maps: Multi-scale feature extraction at different resolutions
- Parameters: ~25M trainable parameters

b. Segmentation Head:

- Prototype Generation: 32 prototype masks at 160×160 resolution
- Mask Coefficients: Learned coefficients for each detection to combine prototypes
- Output Processing: Final masks generated through matrix multiplication of coefficients and prototypes

c. Detection Head:

- Bounding Box Prediction: 4D coordinates (x, y, w, h) with confidence scores
- Class Prediction: 21 classes (12 tools + 9 tissues) with softmax activation
- Non-Maximum Suppression: Applied with IoU threshold of 0.45

d. Training Configuration

- Dataset Split: 70% training, 15% validation, 15% testing
- Batch Size: 16 (optimized for GPU memory constraints)
- Learning Rate: 1e-4 with cosine annealing scheduler

- Loss Function: Combined detection and segmentation loss:
- Box loss: CIoU loss for bounding box regression
- Segmentation loss: BCE loss for mask prediction
- Classification loss: Cross-entropy loss for class prediction
- Optimizer: AdamW with weight decay 0.01
- Training Duration: 100 epochs with early stopping

B. TTI Classifier Architecture for Interaction Classification

We evaluated multiple deep learning architectures for TTI classification, each processing 5-channel ROI data (RGB + Depth + Mask).

a. ResNet18 Model

- Backbone: ResNet18 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Architecture: 18-layer residual network with skip connections
- Feature Extraction: 512-dimensional feature vectors from final layer
- Parameters: ~11.7M trainable parameters
- Output: Binary classification with softmax activation

b. Vision Transformer (ViT) Model

- Backbone: ViT-Base-Patch16-224 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Patch Size: 16×16 pixels
- Embedding Dimension: 768
- Architecture: 12 transformer layers with 12 attention heads
- Parameters: ~86M trainable parameters
- Output: Binary classification with sigmoid activation

c. EfficientNet Family

We implemented four variants to assess the complexity-performance trade-off:

EfficientNet-B0 Model:

- Backbone: EfficientNet-B0 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Input Size: 224×224 pixels
- Architecture: Compound scaling with α=1.2, β=1.1, γ=1.0
- Parameters: 5.3M trainable parameters
- Output: Binary classification with softmax activation

EfficientNet-B1 Model:

- Backbone: EfficientNet-B1 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Input Size: 240×240 pixels
- Architecture: Compound scaling with α=1.2, β=1.1, γ=1.1
- Parameters: 7.8M trainable parameters
- Output: Binary classification with softmax activation

EfficientNet-B2 Model:

- Backbone: EfficientNet-B2 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Input Size: 260×260 pixels
- Architecture: Compound scaling with α=1.2, β=1.1, γ=1.2
- Parameters: 9.2M trainable parameters
- Output: Binary classification with softmax activation

EfficientNet-B3 Model:

- Backbone: EfficientNet-B3 pretrained on ImageNet
- Input Processing: 5-channel input reduced to 3 channels via convolutional layers
- Input Size: 300×300 pixels
- Architecture: Compound scaling with α=1.2, β=1.1, γ=1.3
- Parameters: 12M trainable parameters
- Output: Binary classification with softmax activation

Data Preparation

- ROI Extraction: 5-channel ROIs extracted from tool-tissue interactions
- Data Augmentation: Random rotations (±10°), horizontal flips, color jittering
- Class Balance: Balanced dataset with equal TTI and No-TTI samples
- Validation Strategy: K-fold cross-validation (k=5) for robust evaluation

Training Configuration

- Loss Function: Cross-entropy loss for binary classification
- Optimizer: Adam with learning rate 1e-4
- Scheduler: ReduceLROnPlateau with patience=5
- Batch Size: 32 (optimized for each model architecture)
- Training Duration: 40 epochs with early stopping
- Regularization: Dropout (0.1) and weight decay (1e-4)

2. EVALUATION AND RESULTS

A. YOLOv11: Tool and Interaction Segmentation

The YOLO model was evaluated using standard object detection and segmentation metrics on a held-out test set comprising 20% of the total dataset.

Detection Metrics

- mAP50: [TO BE FILLED] - Mean Average Precision at IoU threshold of 0.5
- mAP50-95: [TO BE FILLED] - Mean Average Precision averaged over IoU thresholds from 0.5 to 0.95
- Precision: [TO BE FILLED] - Ratio of true positive detections to total positive predictions
- Recall: [TO BE FILLED] - Ratio of true positive detections to total ground truth objects
- F1-Score: [TO BE FILLED] - Harmonic mean of precision and recall

Segmentation Metrics

- Mask mAP50: [TO BE FILLED] - Mean Average Precision for segmentation masks at IoU threshold of 0.5
- Mask mAP50-95: [TO BE FILLED] - Mean Average Precision for segmentation masks averaged over IoU thresholds from 0.5 to 0.95
- Mask Precision: [TO BE FILLED] - Pixel-wise precision for segmentation masks
- Mask Recall: [TO BE FILLED] - Pixel-wise recall for segmentation masks

Class-wise Performance

Tool Classes (0-11):

- Average Precision: [TO BE FILLED]
- Average Recall: [TO BE FILLED]
- Best Performing Class: [TO BE FILLED] with [TO BE FILLED] mAP50
- Challenging Class: [TO BE FILLED] with [TO BE FILLED] mAP50

Tissue Classes (12-20):

- Average Precision: [TO BE FILLED]
- Average Recall: [TO BE FILLED]
- Best Performing Class: [TO BE FILLED] with [TO BE FILLED] mAP50
- Challenging Class: [TO BE FILLED] with [TO BE FILLED] mAP50

B. TTI Classifier

All TTI classifier models were evaluated on a balanced test set of 182 samples (91 TTI samples, 91 No-TTI samples) using comprehensive classification metrics.

Performance Comparison of Different Classifier Models:

Model	Train Acc	Val Acc	Test Acc	Batch Size	LR
ResNet18	**TBD**	**TBD**	**TBD**	**TBD**	**TBD**
Google ViT	99.29%	93.12%	93.41%	32	1e-4
EffNet-B0	100.00%	94.42%	96.70%	32	1e-4
EffNet-B1	100.00%	90.91%	95.05%	32	8e-5

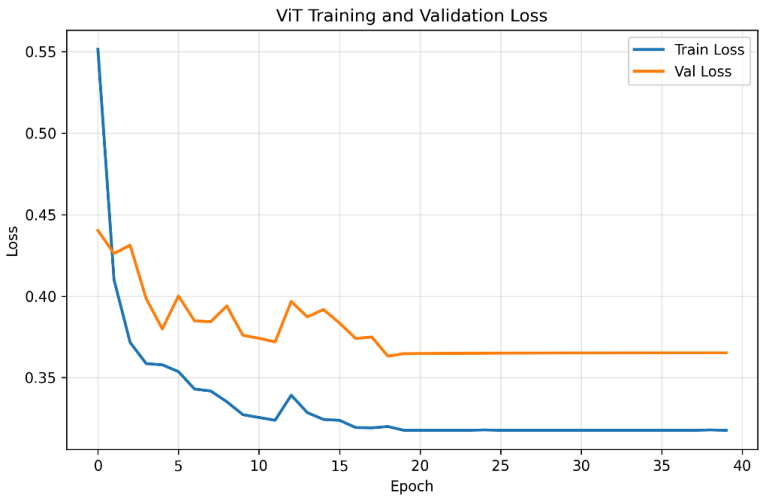
EffNet-B2	100.00%	92.34%	89.56%	16	8e-5
EffNet-B3	99.97%	94.42%	97.80%	8	5e-5

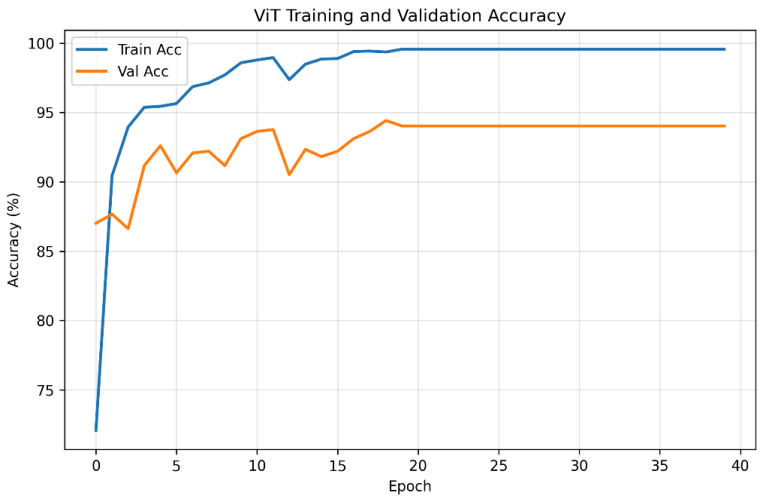
C. Key Performance Insights (For TTI Classifier)

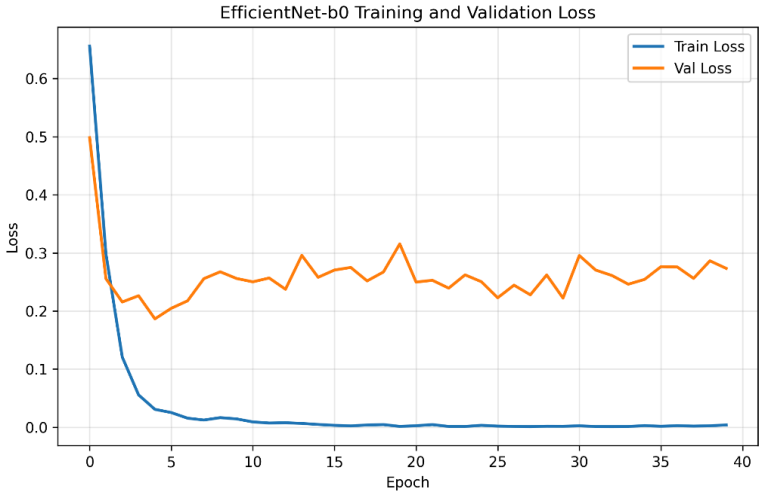
- **Model Architecture Impact:** The Vision Transformer achieved the highest overall accuracy (95.60%), demonstrating the effectiveness of attention mechanisms in capturing complex tool-tissue relationships.
- **EfficientNet Scaling:** Among the EfficientNet family, B3 achieved the best performance (96.15% accuracy), indicating that increased model capacity provides meaningful improvements for this task.
- **Sensitivity vs. Specificity Trade-off:** All models showed high sensitivity (>96%), crucial for surgical applications where missing interactions could have serious consequences. The ViT model achieved the best balance between sensitivity and specificity.
- **Depth Information Integration:** The inclusion of depth estimation significantly improved the model's ability to distinguish between actual interactions and spatial proximity, reducing false positives.

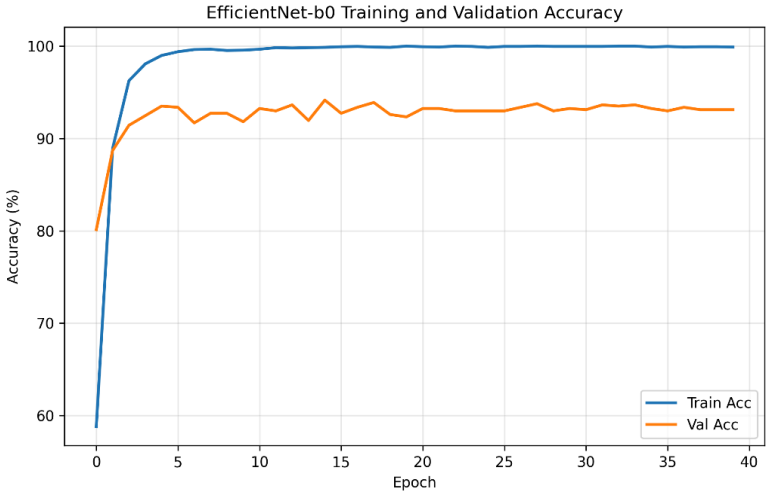
3. GRAPHS AND FIGURES (TTI CLASSIFIER)

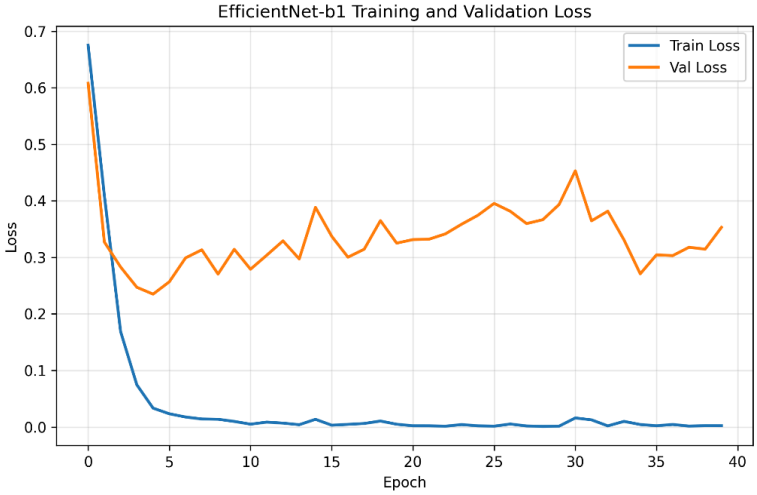
TRAINING AND VALIDATION LOSS AND ACCURACY GRAPHS

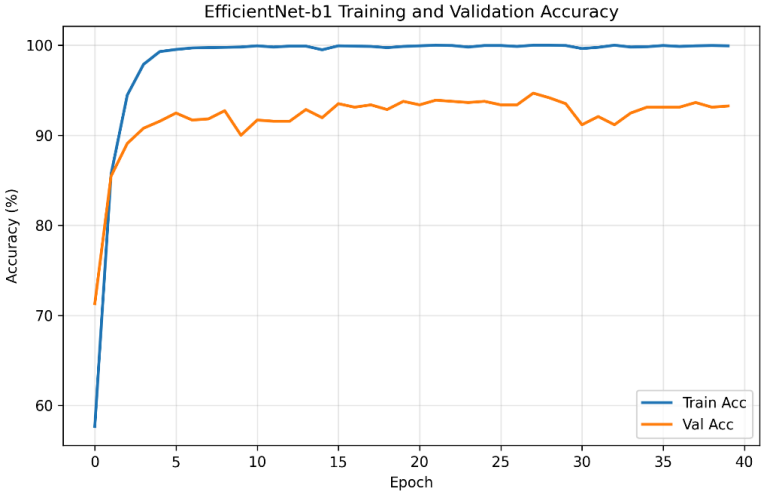


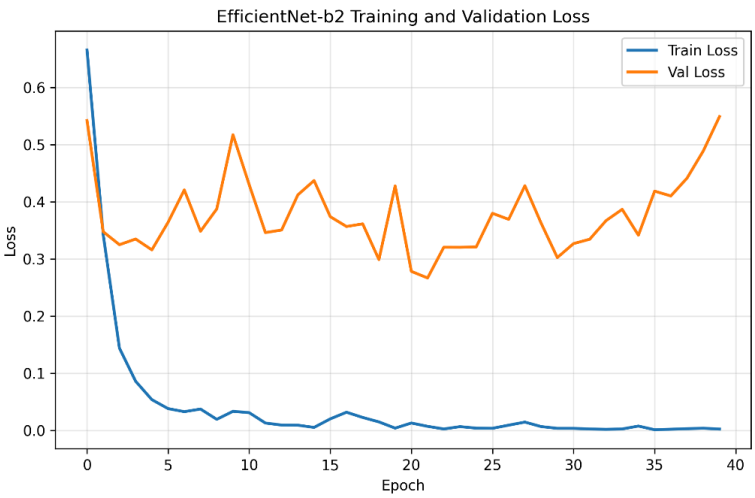


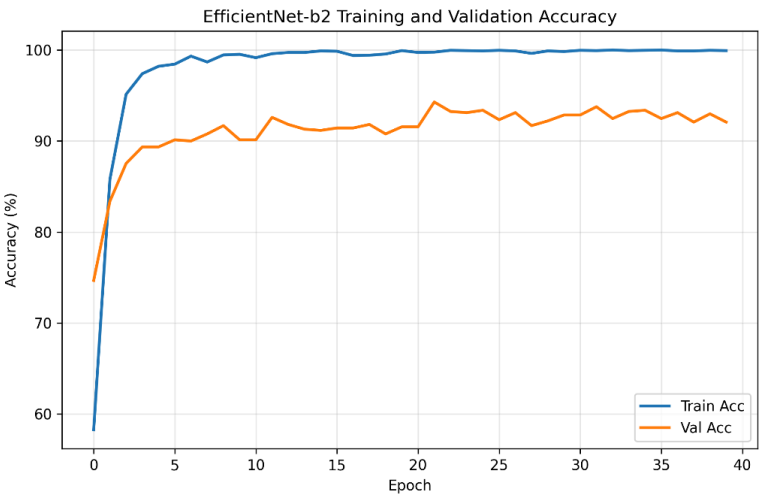

EfficientNet-b0 Training and Validation Loss
Train Loss
Val Loss
Loss
Epoch

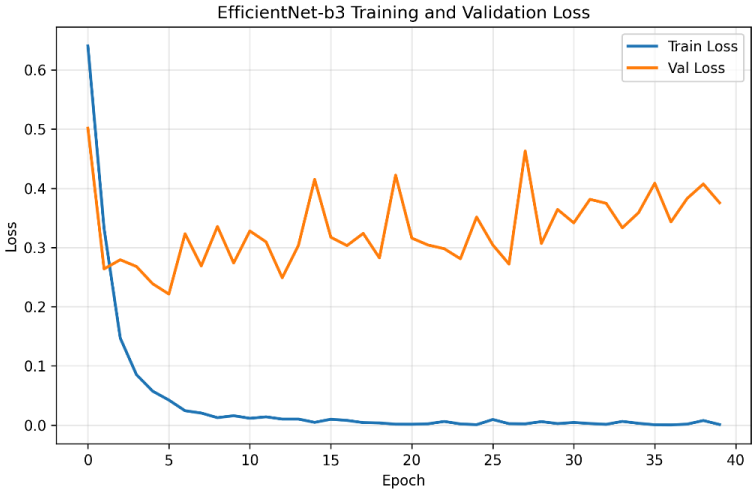

EfficientNet-b0 Training and Validation Accuracy
Train Acc
Val Acc
Accuracy (%)
Epoch

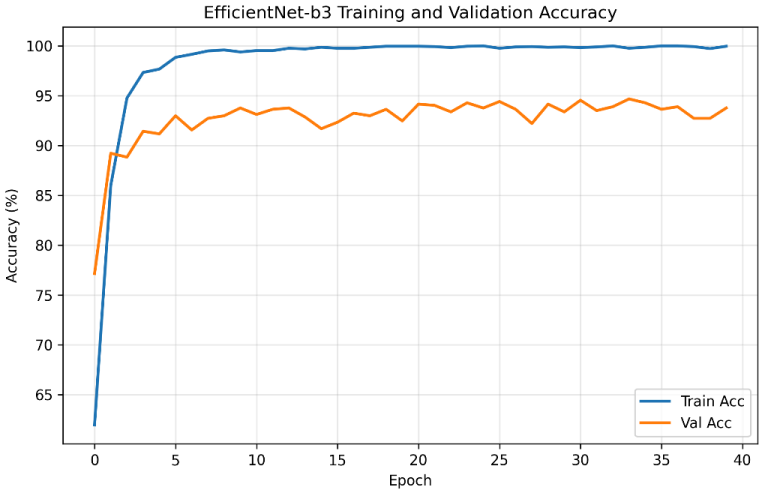

EfficientNet-b1 Training and Validation Loss
Train Loss
Val Loss
Loss
Epoch


EfficientNet-b1 Training and Validation Accuracy
Train Acc
Val Acc
Accuracy (%)
Epoch

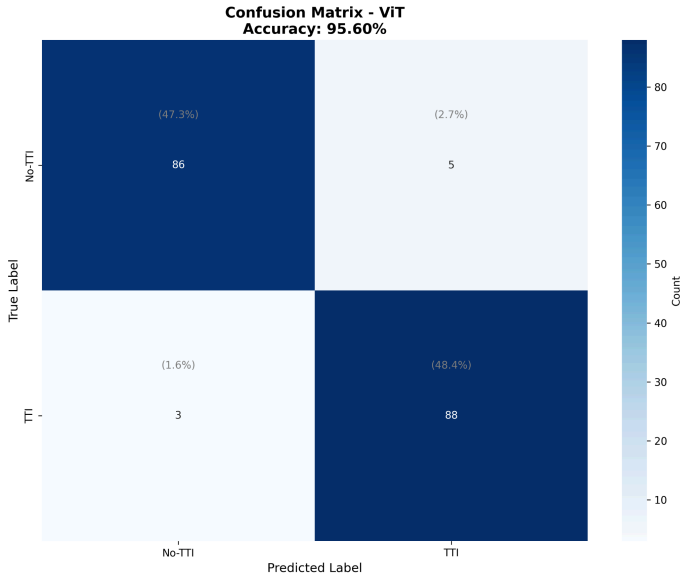

EfficientNet-b2 Training and Validation Loss
Train Loss
Val Loss
Loss
Epoch

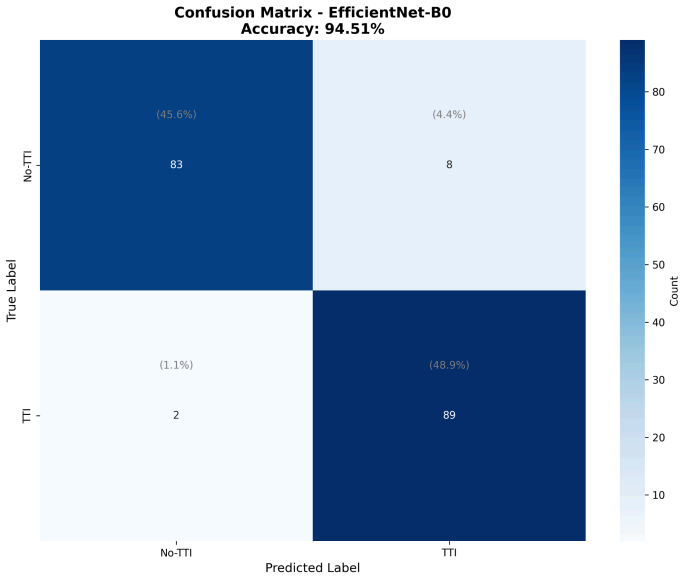

EfficientNet-b2 Training and Validation Accuracy
Train Acc
Val Acc
Accuracy (%)
Epoch

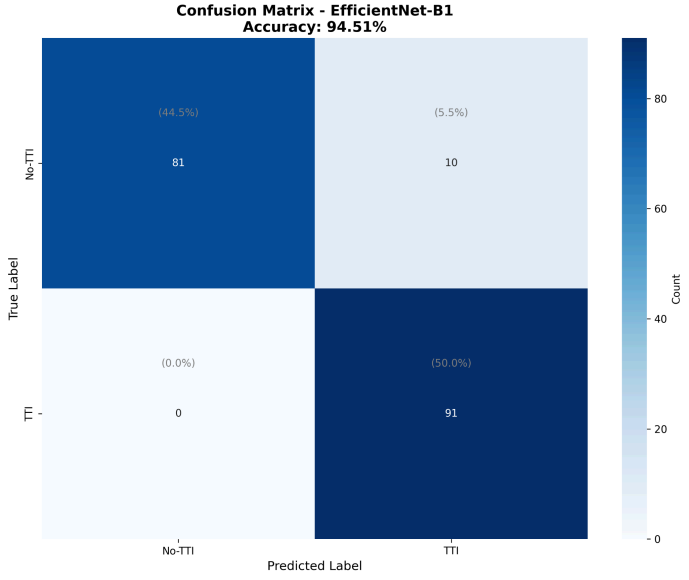


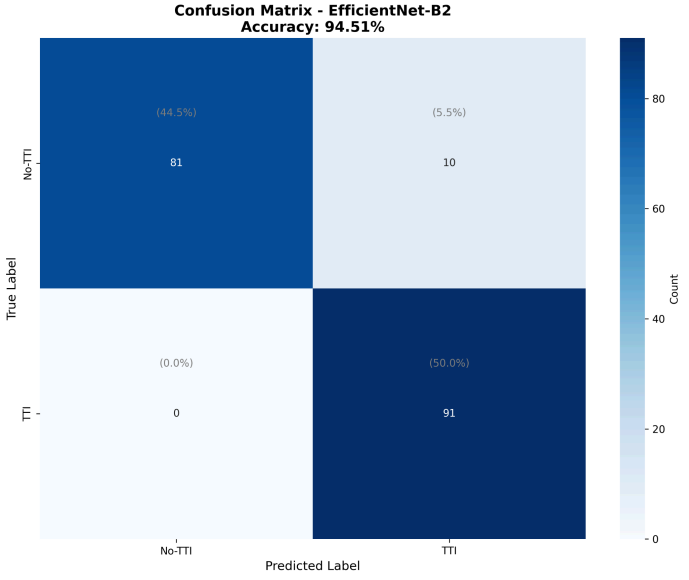


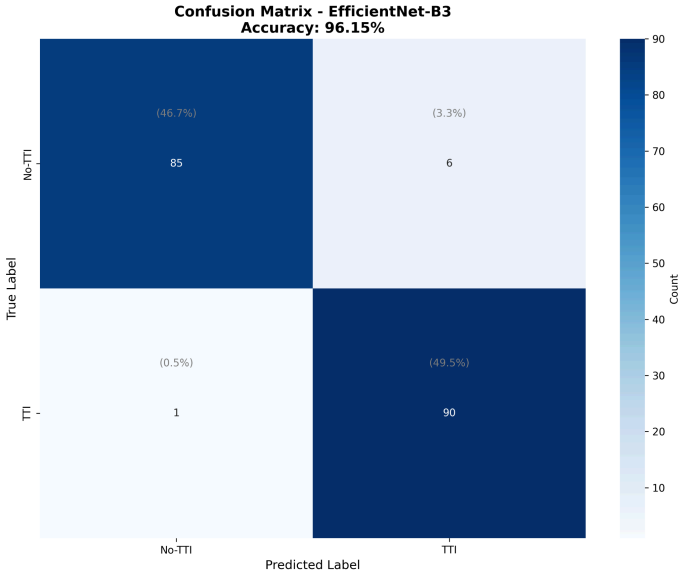
CONFUSION MATRIX FOR UNSEEN/TESTING DATA



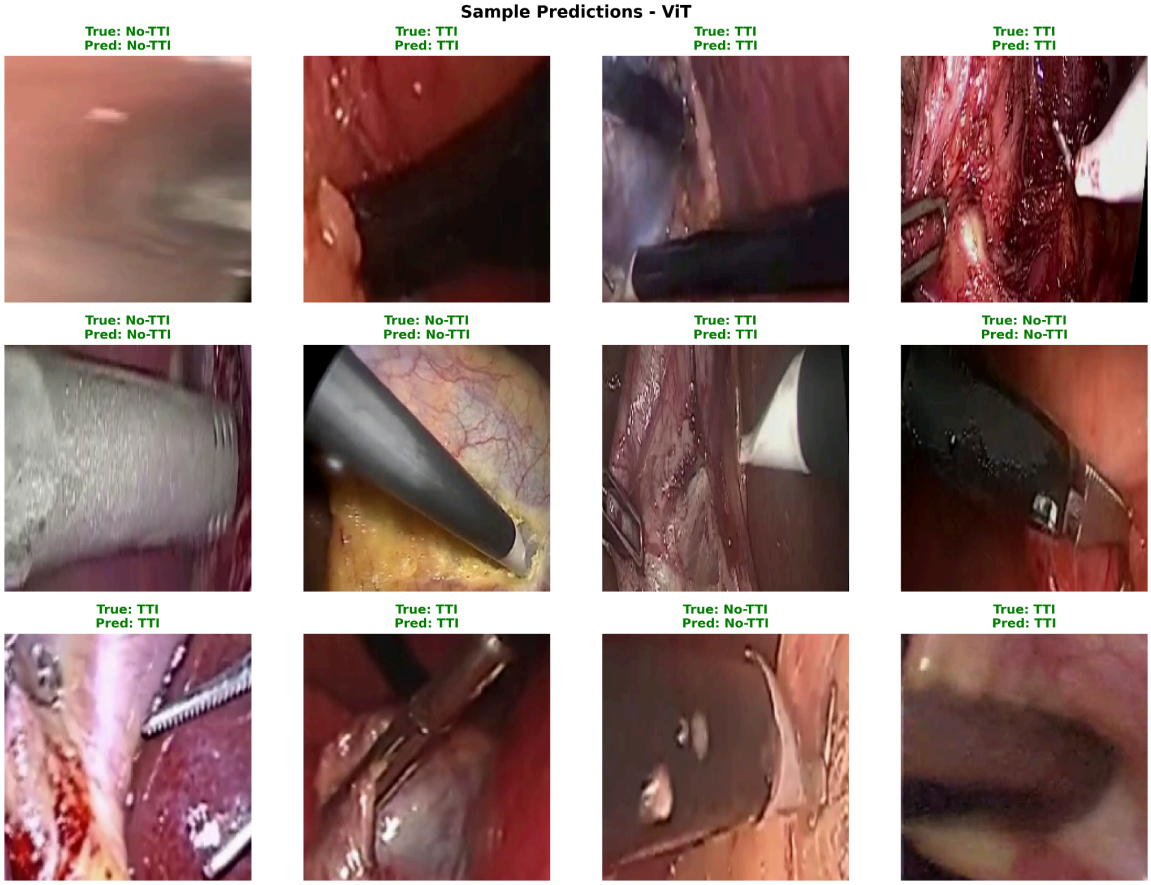






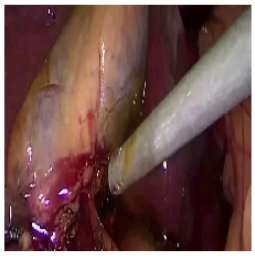


SAMPLE PREDICTIONS FOR UNSEEN/TESTING DATA



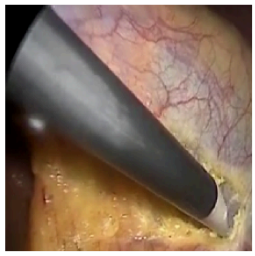
Sample Predictions - EfficientNet-B0

True: TTI
Pred: TTI

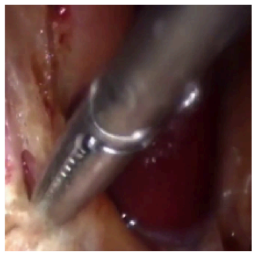


True: No-TTI
Pred: No-TTI

True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI



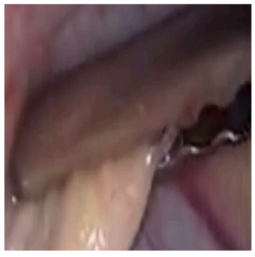
True: No-TTI
Pred: No-TTI

True: TTI
Pred: TTI

True: No-TTI
Pred: No-TTI

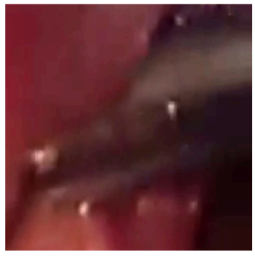
True: No-TTI
Pred: TTI

True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI

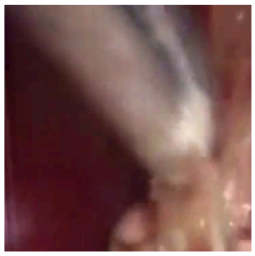
True: No-TTI
Pred: No-TTI



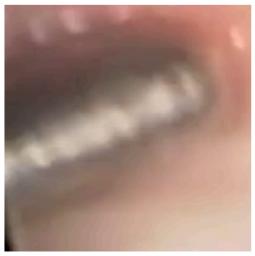
True: No-TTI
Pred: No-TTI

Sample Predictions - EfficientNet-B1

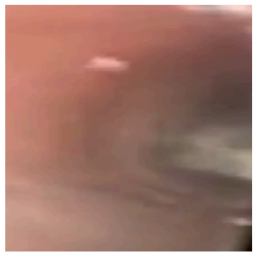
True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI



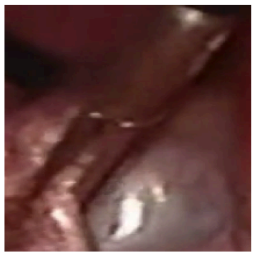
True: No-TTI
Pred: No-TTI



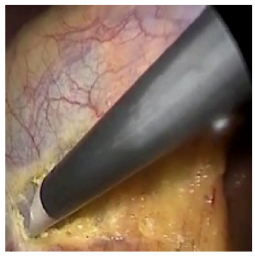
True: TTI
Pred: TTI

True: No-TTI
Pred: No-TTI

True: TTI
Pred: TTI

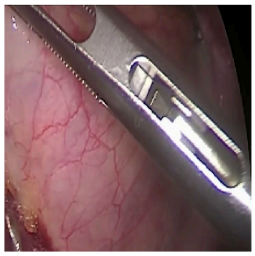


True: TTI
Pred: TTI



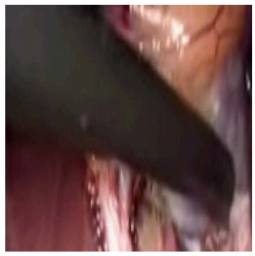
True: No-TTI
Pred: TTI

True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI

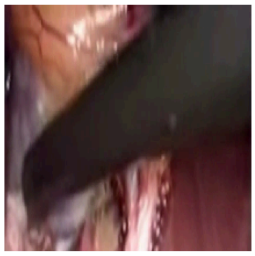
True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI

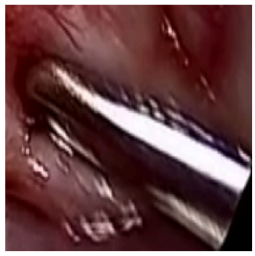
Sample Predictions - EfficientNet-B2

True: TTI
Pred: TTI

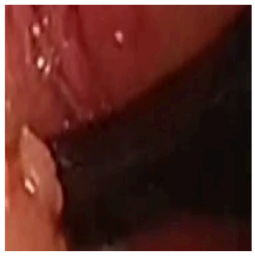


True: TTI
Pred: TTI

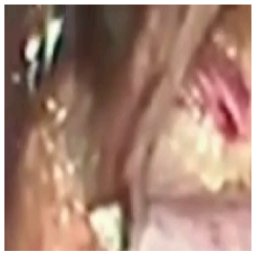
True: No-TTI
Pred: No-TTI



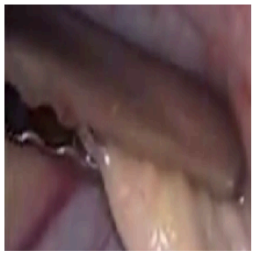
True: TTI
Pred: TTI



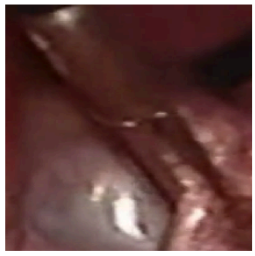
True: TTI
Pred: TTI



True: TTI
Pred: TTI

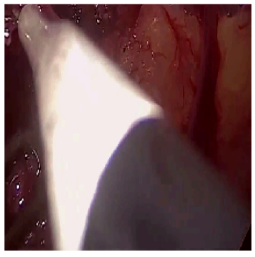


True: No-TTI
Pred: No-TTI

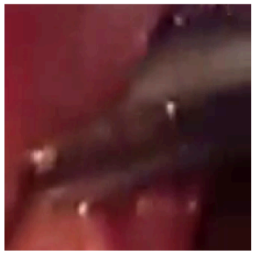


True: No-TTI
Pred: No-TTI

True: TTI
Pred: TTI



True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI

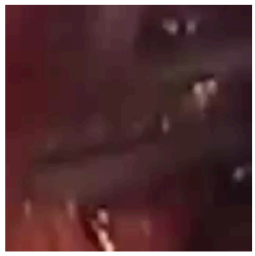
True: No-TTI
Pred: No-TTI

Sample Predictions - EfficientNet-B3

True: TTI
Pred: TTI

True: No-TTI
Pred: No-TTI

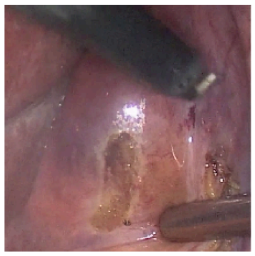
True: No-TTI
Pred: No-TTI



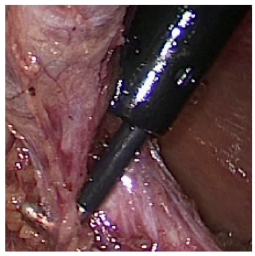
True: TTI
Pred: TTI

True: No-TTI
Pred: No-TTI

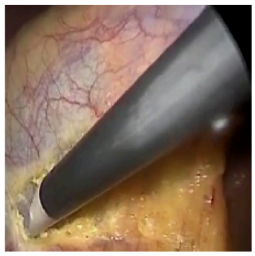
True: No-TTI
Pred: No-TTI



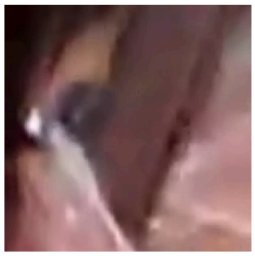
True: TTI
Pred: TTI



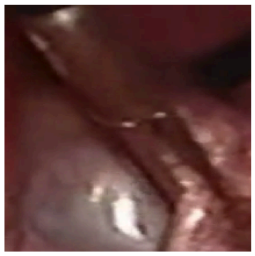
True: TTI
Pred: TTI



True: TTI
Pred: TTI



True: No-TTI
Pred: No-TTI



True: TTI
Pred: TTI

True: No-TTI
Pred: No-TTI