# **Suture Analysis System: Structured Documentation**

## **1. Suture Types**

* **Simple Interrupted**: Individual sutures with separate ends.
* **Continuous**: Single long suture, one start and one end.
* **Purse-string**: Circular stitching pattern.
* **Figure-8**: Crossed stitch resembling the number 8.
* **Subcutaneous**: Placed under the skin surface.

## **2. Classification Labels**

| **Category** | **Labels** | **Example Classes** |
| --- | --- | --- |
| Incision | bent, good, perfect | incision\_bent, incision\_good |
| Knot | good, loose, perfect, tight | knot\_loose, knot\_perfect |
| Suture | good, loose, perfect, tight | suture\_good, suture\_tight |
| Tail | top, end | tail\_top, tail\_end |

## **3. Key Variables & Standardized Measurements**

| **Variable** | **Description** | **Target Value** |
| --- | --- | --- |
| L1 | Incision line to end of stitch | 4 ± 1 mm |
| R1 | Incision to start of knot | 4 ± 1 mm |
| T1, T2 | Tail lengths | 6 ± 3 mm |
| K1 | Incision to center of knot | 6 ± 3 mm |
| α | Stitch angle relative to incision | 90° ± 10° |
| DL1-2 | Left distance between suture points | 4 ± 1 mm |
| DR1-2 | Right distance between suture points | 4 ± 1 mm |

## **4. Preprocessing Techniques**

* Resize to 640x640
* Artifact removal
* Image sharpening
* Heat saturation adjustment
* Noise reduction
* Contour & edge processing

## **5. Model Training Pipeline**

### **Step 1: Preprocess Image**

* Resize, denoise, sharpen
* Convert px to mm (17.5 px = 1 mm)
* Normalize channels

### **Step 2: Object Detection**

* Annotate (Roboflow)
* Train YOLOv8/YOLOv12
* Classify bounding boxes (e.g., knot\_perfect)

### **Step 3: Train Model**

* Use labels and patterns to differentiate suture quality
* Fit using YOLO with defined hyperparameters

### **Step 4: Find Predictions**

* Run inference
* Generate bounding boxes, class IDs, confidence scores

### **Step 5: Fine-Tuning**

* Retrain on difficult examples
* Apply SMOTE/augmentation
* Tune optimizer/loss

### **Optional Step 6: Post-Processing / Scoring**

* Combine measurements with labels
* Apply rule-based or ML scoring (XGBoost, AdaBoost)
* Generate visual reports

## **6. YOLOv8-based Training Hyperparameters**

* Epochs: 300
* Image size: 640
* Batch: 16
* Optimizer: SGD
* Learning Rate: 0.01
* Momentum: 0.937
* Weight Decay: 0.0005
* Augment: TRUE
* Mosaic: 0.2
* Rect: TRUE

## **7. Performance Benchmarks**

### **a. Object Detection Metrics**

| **Metric** | **Description** | **Tool** |
| --- | --- | --- |
| mAP@0.5/0.95 | Detection precision over thresholds | YOLO eval, Roboflow |
| Precision | TP / (TP + FP) | Scikit-learn |
| Recall | TP / (TP + FN) | Scikit-learn |
| F1 Score | Harmonic mean of precision and recall | Scikit-learn |
| Inference Time | Time per image/frame | YOLO logs |

### **b. Measurement Accuracy**

| **Metric** | **Goal** |
| --- | --- |
| Tail length error | < ± 0.5 mm |
| Knot-to-incision distance | < ± 1 mm |
| Suture angle error | < ± 5 degrees |

### **c. Class Prediction Quality**

| **Metric** | **Target** |
| --- | --- |
| Accuracy | > 90% |
| Confusion Matrix | Balanced |
| SMOTE Impact | FN reduction |

### **d. Ablation Studies**

| **Change** | **Result** |
| --- | --- |
| Sharpening | mAP ↑, tail error ↓ |
| Pixel-to-mm calibration | Length error ↓ |
| SGD → Adam | Faster convergence, F1 ↑ |
| SMOTE balancing | Recall ↑ |

### **e. Final Scoring Output**

* Score agreement with experts
* Feedback clarity
* Cohen's Kappa for rater reliability

## **8. Additional Processing**

* Roboflow for annotation
* Object-oriented bounding box strategy
* Custom YAML label files
* Fine-tune at 100-300 epochs

## **9. Optimization Techniques**

### **Recommended for This System:**

| **Stage** | **Method** | **Why** |
| --- | --- | --- |
| YOLOv8 Training | SGD (with momentum) | Efficient and scalable for image tasks |
| Fine-tuning | Adam | Faster convergence, good for edge cases |
| Post-YOLO Scoring | XGBoost / AdaBoost | Good on structured tabular features |
| Hyperparameter Tuning | Random Search + Early Stopping | Efficient tuning strategy |
| Custom Scoring (Optional) | Evolutionary Algorithm | Experimental tuning of rule weights |

## **10. Visual Benchmark Enhancements**

* Superglued scale for camera calibration
* Skin scribe for incision clarity
* Rule to ignore incomplete sutures (e.g., bottom one cut off)

## **11. Suggested Enhancements**

* Automated incision detection/extrapolation
* Feature extractor for LE, R, L, K1, T1/T2, α, D
* Outlier detector for occluded/missing components

## **12. Project Directory Structure**

| /data  /labeled*\_data\_*all/  /data\_obb/  /enhanced*\_images\_*all/  /new\_ignore/  /raw\_data/  /isolated\_sutures/  /runs  /detect/train/exp1  /segment/train/exp1  /classify/train |
| --- |

## **13. YOLOv8 Training Configs**

| **# Basic Detection** model: yolov8n.pt data: data/labeled\_data/data.yaml epochs: 100 batch: 16  **# OBB Detection** model: yolov8s.pt data: data/data\_obb/data.yaml epochs: 150 batch: 32  **# Enhanced Model** model: yolov8m.pt data: data/new\_ignore/data.yaml epochs: 200 batch: 24 |
| --- |

## **14. Evaluation Tools**

| **# Analyze training run** python tools/analyze\_results.py runs/detect/train/exp1  **# Compare runs** python tools/compare\_runs.py runs/detect/train/exp1 runs/detect/train/exp2 |
| --- |

## **15. ML Workflow and Pipeline Documentation**

### **Overview**

This document outlines the machine learning workflow and pipeline for the suture analysis project, detailing each component and its functionality.

### **1. Model Architecture**

* **Base Model**: YOLOv8 for object detection
* **Additional Components**:  
  + Enhanced preprocessing pipeline
  + XGBoost classifier for quality assessment
  + Automated scale calibration system

### **2. Data Pipeline**

#### **Preprocessing Steps**

* RGB Color Space Conversion
* CLAHE Enhancement:  
  + Adaptive histogram equalization
  + Contrast enhancement optimization
* Bilateral Filtering:  
  + Noise reduction
  + Edge preservation
* Image Normalization (Optional)
* Resizing to 640x640 pixels

### **3. Training Pipeline**

#### **Data Organization**

| dataset/ ├── train/ │ ├── images/ │ └── labels/ ├── val/ │ ├── images/ │ └── labels/ └── test/  ├── images/  └── labels/ |
| --- |

#### **Training Configuration**

* **Batch Size**: 16
* **Epochs**: 100-300
* **Optimizer**: AdamW
* **Learning Rate**: Cosine Annealing Schedule
* **Augmentation**: Test-time augmentation enabled
* **Advanced Features**:  
  + Model ensemble support
  + GPU acceleration
  + Early stopping with patience

### **4. Inference Pipeline**

#### **Detection Phase**

1. Image Preprocessing  
   * Apply CLAHE and bilateral filtering
   * Normalize and resize
2. YOLOv8 Inference  
   * Confidence thresholding
   * Non-Maximum Suppression (NMS)

#### **Analysis Phase**

* **Measurements**:  
  + Stitch length calculation
  + Angle analysis
  + Pattern symmetry evaluation
  + Spacing uniformity assessment
  + Depth consistency verification

### **5. Quality Assessment**

#### **Feature Extraction**

* **Geometric Features**:  
  + Mean stitch length
  + Standard deviation of lengths
  + Mean angle deviation
  + Angle consistency
* **Pattern Features**:  
  + Symmetry scores
  + Spacing uniformity
  + Depth consistency
* **Quality Metrics**:  
  + Pattern regularity
  + Stitch tension
  + Overall consistency

#### **Classification**

* **Primary**: XGBoost Model  
  + Input: Extracted features
  + Output: Quality classification
  + Classes: good, tight, loose
* **Backup**: Rule-based System  
  + Fallback for edge cases
  + Threshold-based classification

### **6. Performance Metrics**

#### **Model Evaluation**

* Precision
* Recall
* mean Average Precision (mAP)
* F1 Score

#### **Feature Analysis**

* Ablation studies
* Feature importance ranking
* Performance breakdown by class

### **7. Optimization Features**

#### **Runtime Optimization**

* GPU acceleration
* Batch processing
* Test-time augmentation
* Model ensemble inference

#### **Quality Assurance**

* Automated error logging
* Scale calibration
* Validation checks
* Performance monitoring

### **8. Future Improvements**

* Automated scale detection enhancements
* Advanced artifact removal
* Image quality optimization
* Real-time processing capabilities

#### Note: This documentation provides a complete, optimized view of the Suture Analysis System—from image preprocessing to training, benchmarking, and advanced post-analysis scoring.