Complete Implementation Guide

Surgical Duration Prediction with Deep Learning

Part 1: The Problem We Are Solving

What is the Task?

Given a video of a laparoscopic cholecystectomy (gallbladder removal surgery), we want to predict at any moment:

1. How much longer will the current phase last? (phase remaining time)
2. How much longer will the entire surgery last? (surgery remaining time)
3. Which phase are we in right now? (phase classification)

Why Is This Hard?

- Surgeries vary wildly: One cholecystectomy might take 25 minutes, another 90 minutes
- Same visual appearance, different times: A grasper tool could mean 5 min or 40 min remaining depending on context
- Single frames are ambiguous: You need to understand the sequence of events over time

Clinical Value

- OR schedulers: Know when to prepare next patient
- Surgical team: Anticipate upcoming phases
- Anesthesiologists: Plan medication timing
- Hospital admin: Optimize resource allocation

Part 2: Our Architecture - CNN + LSTM

The Big Picture

Our model has two main components working together:

1. CNN (ResNet-50): Looks at each video frame and extracts visual features. It answers "what is in this image?" by converting raw pixels into meaningful representations.

2. LSTM: Processes the sequence of features over time. It answers "given what I have seen so far, what comes next?" by maintaining memory of past frames.

Data Flow Through the Model

```
Video Frames [B, 30, 3, 224, 224]
        |
        v
    ResNet-50 (CNN) --> extracts 2048-dim features per frame
        |
        v
Features [B, 30, 2048] + Elapsed Time [B, 30, 1]
        |
        v (concatenate)
Combined [B, 30, 2049]
        |
        v
    LSTM (256 hidden) --> temporal reasoning
        |
        v
    Hidden State [B, 256]
        |
```

```
      v
  4 Prediction Heads:
  - Phase classification (7 classes)
  - Phase time remaining (regression)
  - Surgery time remaining (regression)
  - Progress 0-1 (regression)
```

Why ResNet-50?

- Deep enough: 50 layers can learn complex hierarchical features
- Residual connections: Solves vanishing gradient problem in deep networks
- Pretrained on ImageNet: Already knows edges, textures, shapes, objects from 1.2M images
- Transfer learning: General visual knowledge transfers well to surgical images

Why LSTM?

- Temporal dependencies: Surgery has structure - phases follow specific order
- Memory: Can remember what happened earlier in the sequence
- Context: "10 minutes of dissection" is more informative than a single frame
- Sequential processing: Naturally handles variable-length sequences

Why Multi-Task Learning?

We train ONE model to predict FOUR things simultaneously. Benefits:

1. Tasks help each other: Phase recognition helps time prediction
2. Shared representation: CNN+LSTM learns features useful for all tasks
3. Regularization: Each task constrains learning, preventing overfitting
4. Efficiency: One forward pass produces multiple outputs

Part 3: Key Papers and Their Contributions

1. EndoNet (Twinanda et al., IEEE-TMI 2016)

Title: "EndoNet: A Deep Architecture for Recognition Tasks on Laparoscopic Videos"

Key contributions:
- First CNN for multi-task learning on surgical videos
- Introduced Cholec80 dataset (80 cholecystectomy videos)
- Multi-task: phase recognition + tool detection together

What we used: Multi-task architecture concept, dataset structure, phase definitions

2. Aksamentov et al. (MICCAI 2017)

Title: "Deep Neural Networks Predict Remaining Surgery Duration from Cholecystectomy Videos"

Key contributions:
- CNN + LSTM architecture for time prediction
- Elapsed time as input to LSTM (crucial insight!)
- L1 loss for regression (robust to outliers)
- Achieved MAE of 7.7 minutes on Cholec120

What we used: Elapsed time input, L1 loss, CNN-LSTM pipeline architecture

3. RSDNet (Twinanda et al., IEEE-TMI 2019)

Title: "RSDNet: Learning to Predict Remaining Surgery Duration Without Manual Annotations"

Key contributions:
- Progress signal as self-supervised learning (no annotation needed!)
- Progress = elapsed_time / total_duration (free from timestamps)
- Learning progress helps learn time prediction
- Achieved MAE of 8.1 minutes on Cholec120

What we used: Progress prediction head, self-supervised signal concept

4. Less is More (Yengera et al., 2018)

Title: "Surgical Phase Recognition with Less Annotations through Self-Supervised Pre-training"

Key contributions:
- RSD prediction as pre-training task
- End-to-end CNN-LSTM training
- Shows how progress understanding transfers to phase recognition

What we used: Concept of progress as auxiliary task that helps main task

Part 4: Implementation Details

Image Preprocessing

1. Resize to 224x224 (ResNet standard input size)
2. ImageNet normalization: mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225]
3. Training augmentation: RandomCrop, HorizontalFlip, ColorJitter
4. No augmentation for validation (deterministic evaluation)

Surgical Duration Prediction - Deep Learning Guide

Loss Function

```
total_loss = 0.3 * phase_loss        # CrossEntropy for classification
         + 0.2 * phase_time_loss   # L1 for regression
         + 0.3 * surgery_time_loss # L1 for regression (main goal)
         + 0.2 * progress_loss     # L1 for regression (self-supervised)
```

Training Configuration

- Optimizer: Adam with different learning rates
  - LSTM and heads: lr = 1e-4 (learning from scratch)
  - CNN layer4: lr = 1e-5 (fine-tuning pretrained weights)
- Batch size: 4 (limited by GPU memory for video sequences)
- Early stopping: patience = 3 epochs
- Sequence length: 30 frames (30 seconds at 1fps)

Freezing Strategy

Frozen CNN: All ResNet layers frozen, only LSTM trains
- Fast training, uses ImageNet features as-is
- Good baseline, but may not capture surgical-specific features

Unfrozen Layer4: Early layers frozen, layer4 trainable
- Layer4 adapts to surgical domain
- Preserves low-level features (edges, textures)
- Expected to perform better on surgical data

Part 5: Benchmark Results

Published results on Cholec120 dataset for comparison:

- Aksamentov et al. (2017): MAE = 7.7 minutes
- RSDNet (2019): MAE = 8.1 minutes
- TransLocal (2024): MAE = 7.1 minutes

Target: Getting close to 7-8 minutes MAE would be competitive with state-of-the-art published results.