Online Materials for Efficient Surgical Tool Recognition via HMM-Stabilized Deep Learning

Haifeng Wang, Hao Xu, Jun Wang, Jian Zhou, Ke Deng

I. EXISTING METHODS FOR SURGICAL VIDEO ANALYSIS

Figure 1 shows the architecture of ToolNet, PhaseNet, EndoNet, SwinNet and some of their extensions equipped with LSTM and attention mechanism [1], [2], [3], [4], [5].

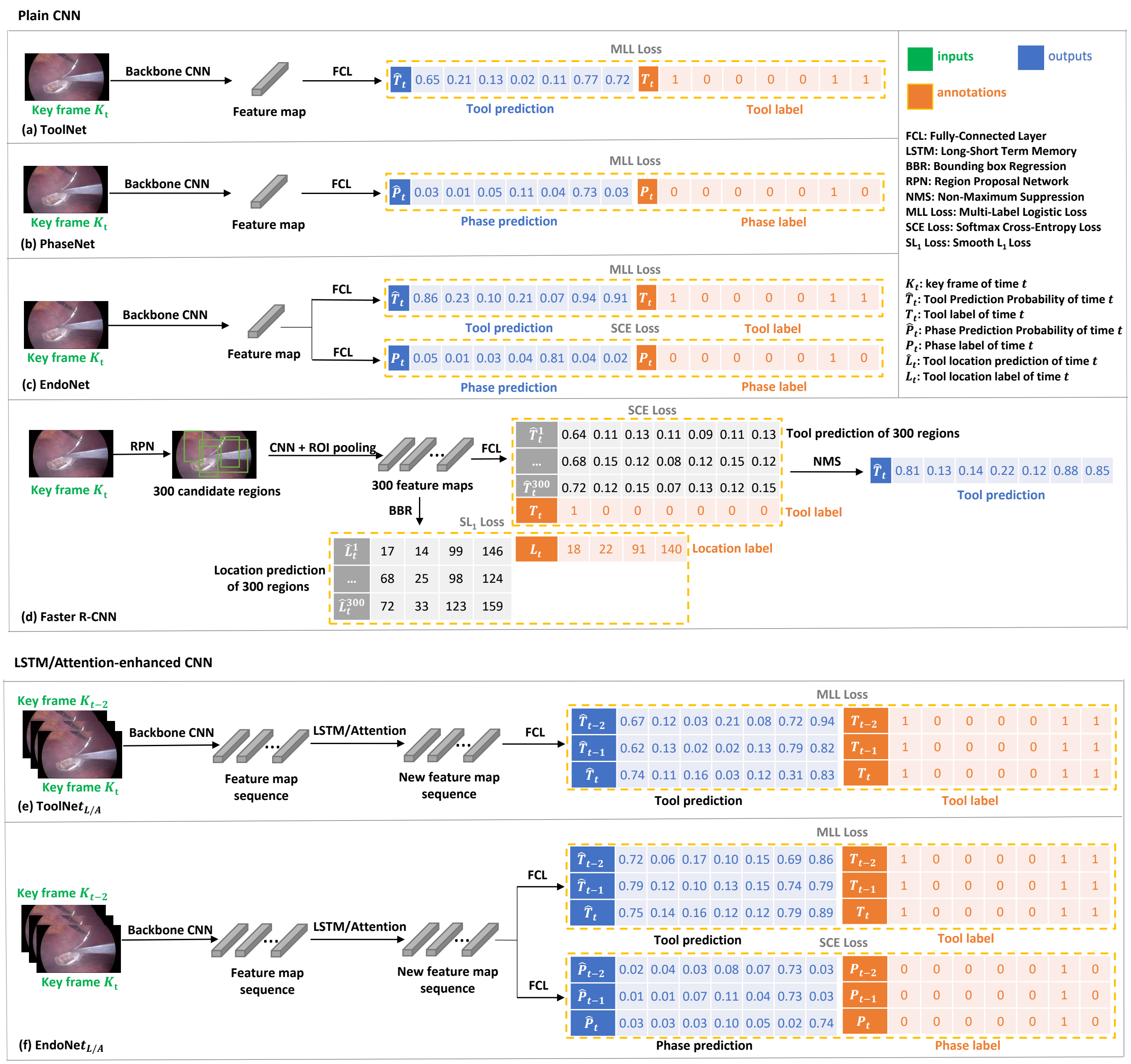

Fig. 1. Graphical illustration of deep-learning methods for surgical tool recognition.

II. Detailed Calculations for Inferring HMM-stabilized Deep Learning

A. Parameter Estimation via the EM Algorithm

In principle, the parameters of HMM-stabilized deep learning model in can be estimated via the maximum likelihood principle, i.e., maximizing the likelihood, which is a function of both true and predicted tool labels with respect to $\boldsymbol{\theta}$. Because, true phase and tool labels are observed for training data only, we typically rely on the expectation-maximization algorithm [6] to do the optimization, treating the unobserved tool labels as missing data. The Q-function of E-step is shown as follows:

$$Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)}) = \sum_{i=1}^{m}\sum_{\mathcal{I}_i} \log \mathbb{P}(\mathcal{I}_i)\mathbb{P}\left(\mathcal{I}_i|\mathcal{I}_i^{obs}, \boldsymbol{\theta}^{(s)}\right), \tag{1}$$

where $\boldsymbol{\theta}^{(s)}$ is the estimation of model parameter $\boldsymbol{\theta}$ in the s-th iteration of the EM algorithm.

After substituting $\mathbb{P}(\mathcal{I}_i)$ into Eq. (1) we obtain

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)}) =& \sum_{i=1}^{m}\sum_{\mathcal{I}_i} \log\left(\textbf{Multinomial}(P_{i,1}|\boldsymbol{\alpha})\cdot\prod_{t=2}^{n_i}\mathbf{A}(P_{i,t-1},P_{i,t})\right)\mathbb{P}\left(\mathcal{I}_i|\mathcal{I}_i^{obs}, \boldsymbol{\theta}^{(s)}\right)\\
&+\sum_{i=1}^{m}\sum_{\mathcal{I}_i} \log\left(\prod_{\tau\in\mathcal{T}}\textbf{Bernoulli}(T_{i,1,\tau}|\beta_{\tau,P_{i,1}})\cdot\prod_{t=2}^{n_i}\mathbf{A}_{\tau,P_{i,t}}(T_{i,t-1,\tau},T_{i,t,\tau})\right)\mathbb{P}\left(\mathcal{I}_i|\mathcal{I}_i^{obs}, \boldsymbol{\theta}^{(s)}\right)\\
&+\sum_{i=1}^{m}\sum_{\mathcal{I}_i} \log\left(\prod_{t=1}^{n_i}\mathbf{B}(P_{i,t},\hat{P}_{i,t})\right)\mathbb{P}\left(\mathcal{I}_i|\mathcal{I}_i^{obs}, \boldsymbol{\theta}^{(s)}\right)\\
&+\sum_{i=1}^{m}\sum_{\mathcal{I}_i} \log\left(\prod_{\tau\in\mathcal{T}}\prod_{t=1}^{n_i}\mathbf{B}_{\tau}(T_{i,t,\tau},\hat{T}_{i,t,\tau})\right)\mathbb{P}\left(\mathcal{I}_i|\mathcal{I}_i^{obs}, \boldsymbol{\theta}^{(s)}\right).
\end{aligned}\tag{2}$$

By further grouping similar terms in Eq. (2) based on the parameters $\boldsymbol{\theta} = (\boldsymbol{\alpha}, \mathbf{A}; \boldsymbol{\beta}, \mathcal{A}; \mathbf{B}, \mathcal{B})$, we have

$$\begin{aligned}
Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)}) =& \sum_{\varrho\in\mathcal{P}} \log\alpha_{\varrho}\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]\\
&+\sum_{\varrho_i\in\mathcal{P}}\sum_{\varrho_j\in\mathcal{P}}\log\mathbf{A}(\varrho_i,\varrho_j)\mathbb{E}\left[\mathbb{N}(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]\\
&+\sum_{\tau\in\mathcal{T}}\sum_{l=0}^{1}((1-l)\log(1-\beta_{\tau,\varrho})+l\log(\beta_{\tau,\varrho}))\mathbb{E}\left[\mathbb{N}(T_{\cdot,1,\tau}=l,P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]\\
&+\sum_{\tau\in\mathcal{T}}\sum_{\varrho\in\mathcal{P}}\sum_{m=0}^{1}\sum_{n=0}^{1}\log\mathbf{A}_{\tau,\varrho}(m,n)\mathbb{E}\left[\mathbb{N}(T_{\cdot,t-1,\tau}=m,T_{\cdot,t,\tau}=n,P_{\cdot,t}=\varrho|\boldsymbol{\theta}^{(s)})\right]\}\\
&+\sum_{\varrho_i\in\mathcal{P}}\sum_{\varrho_j\in\mathcal{P}}\log\mathbf{B}(\varrho_i,\varrho_j)\mathbb{E}\left[\mathbb{N}(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]\\
&+\sum_{\tau\in\mathcal{T}}\sum_{m=0}^{1}\sum_{n=0}^{1}\log\mathbf{B}_{\tau}(m,n)\mathbb{E}\left[\mathbb{N}(T_{\cdot,t,\tau}=m,\hat{T}_{\cdot,t,\tau}=n|\boldsymbol{\theta}^{(s)})\right],
\end{aligned}\tag{3}$$

where

$$\begin{aligned}
\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right] &= \sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,1}=\varrho)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\
\mathbb{E}\left[\mathbb{N}(T_{\cdot,1,\tau}=j,P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right] &= \sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,1,\tau}=j,P_{i,1}=\varrho)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\
\mathbb{E}\left[\mathbb{N}(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right] &= \sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,t-1}=\varrho_i,P_{i,t}=\varrho_j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\
\mathbb{E}\left[\mathbb{N}(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right] &= \sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,t}=\varrho_i,\hat{P}_{i,t}=\varrho_j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\
\mathbb{E}\left[\mathbb{N}(T_{\cdot,t-1,\tau}=i,T_{\cdot,t,\tau}=j,P_{\cdot,t}=\varrho|\boldsymbol{\theta}^{(s)})\right] &= \sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,t-1,\tau}=i,T_{i,t,\tau}=j,P_{i,t}=\varrho)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),
\end{aligned}$$

$$\mathbb{E}\left[\mathbb{N}(T_{\cdot,t,\tau}=i,\hat{T}_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right]=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,t,\tau}=i,\hat{T}_{i,t,\tau}=j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}).$$

To identify the value of $\boldsymbol{\theta}$ that maximizes the Q-function, we solve the equation by setting the derivatives of the Q-function with respect to $\boldsymbol{\theta}$ to zero in M-step. Considering $\sum_{\varrho\in\mathcal{P}}\alpha_\varrho=1$, we apply the Lagrange multiplier method to determine the optimal value of $\{\alpha_\varrho\}_{\varrho\in\mathcal{P}}$, resulting in the following equation,

$$\frac{\partial Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)})-\lambda(\sum_{\varrho\in\mathcal{P}}\alpha_\varrho-1)}{\partial\alpha_\varrho}=0,(\varrho\in\mathcal{P}). \tag{4}$$

Eq. (4) is simplified as follows,

$$\frac{\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]}{\alpha_\varrho}-\lambda=0,(\varrho\in\mathcal{P}). \tag{5}$$

After taking the sum of Eq. (5) over all $\varrho\in\mathcal{P}$, we have

$$\sum_{\varrho\in\mathcal{P}}\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]=\sum_{\varrho\in\mathcal{P}}\lambda\alpha_\varrho=\lambda. \tag{6}$$

When we substitute Eq. (6) into Eq. (5), we derive the M-step updating equation for the parameters $\boldsymbol{\alpha}=\{\alpha_\varrho\}_{\varrho\in\mathcal{P}}$. By applying the similar Lagrange multiplier method to all parameters, we come up with the following updating function to iteratively improve the estimation of $\boldsymbol{\theta}$:

$$\alpha_\varrho^{(s+1)}=\frac{\mathbb{E}\left[(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]}{\sum_{\varrho'=1}^{L}\mathbb{E}\left[(P_{\cdot,1}=\varrho'|\boldsymbol{\theta}^{(s)})\right]}, \tag{7}$$

$$\beta_{\tau,\varrho}^{(s+1)}=\frac{\mathbb{E}\left[(T_{\cdot,1,\tau}=1,P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(n)})\right]}{\sum_{j'=0}^{1}\mathbb{E}\left[(T_{\cdot,1,\tau}=j',P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]}, \tag{8}$$

$$A^{(s+1)}(\varrho_i,\varrho_j)=\frac{\mathbb{E}\left[(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]}{\sum_{\varrho_{j'}=1}^{L}\mathbb{E}\left[(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_{j'}|\boldsymbol{\theta}^{(s)})\right]}, \tag{9}$$

$$B^{(s+1)}(\varrho_i,\varrho_j)=\frac{\mathbb{E}\left[(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]}{\sum_{\varrho_{j'}=1}^{L}\mathbb{E}\left[(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_{j'}|\boldsymbol{\theta}^{(s)})\right]}, \tag{10}$$

$$A_{\tau,\varrho}^{(s+1)}(i,j)=\frac{\mathbb{E}\left[(T_{\cdot,t-1,\tau}=i,T_{\cdot,t,\tau}=j,P_{\cdot,t}=\varrho|\boldsymbol{\theta}^{(s)})\right]}{\sum_{j'=0}^{1}\mathbb{E}\left[(T_{\cdot,t-1,\tau}=i,T_{\cdot,t,\tau}=j',P_{\cdot,t}=\varrho|\boldsymbol{\theta}^{(s)})\right]}, \tag{11}$$

$$B_{\tau}^{(s+1)}(i,j)=\frac{\mathbb{E}\left[(T_{\cdot,t,\tau}=i,\hat{T}_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right]}{\sum_{j'=0}^{1}\mathbb{E}\left[(T_{\cdot,t,\tau}=i,\hat{T}_{\cdot,t,\tau}=j'|\boldsymbol{\theta}^{(s)})\right]}, \tag{12}$$

Direct calculation based on the above formula is clearly forbidden because it involves the enumeration of all possible values of $\mathcal{I}_i$, whose complexity increases exponentially with the length of video $\mathbf{V}_i$. In practice, efficient computation with linear complexity can be achieved by following the standard Baum-Welch algorithm [7]. We employ the standard forward-backward algorithm to compute the six expectations mentioned in Eq. (3), given the complexity of enumerating hidden states (refer to Appendix II-B for more details). Considering that the iterative estimation of the Baum-Welch algorithm is computationally expensive, we can also take a shortcut to estimate some of the parameters directly based on the training data only. For example, the phase transition matrices $\mathbf{A}$ can be conveniently estimated by the empirical transition matrices calculated from the observed phase labels in the training data. The start probability $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ can be estimated by the corresponding empirical frequencies. Such a strategy is computationally convenient with little loss of estimation efficiency when videos with and without training labels have similar transition patterns.

B. Fast Computation of the E-Step

In the parameter estimation steps of HMM-stabilized methods, we use forward-backward algorithm to get the expectation of interest in E-step. Due to the complexity of hidden state enumeration, we utilize the standard forward-backward algorithm to calculate the expectation in Eq. (3) via EM algorithm.

For the observation $(\hat{\mathbf{P}}_i, \hat{\mathbf{T}}_i)$ of n_i key frames in video i and parameters $\boldsymbol{\theta}^{(t)}$ of the HMM-stabilized model, we define the forward and backward variables as follows,

$$\mathbb{U}_{i,t}(\varrho, \tau) = \mathbb{P}(\hat{\mathbf{P}}_{i,[n\le t]}, \hat{\mathbf{T}}_{i,[n\le t]}, P_{i,t} = \varrho, T_{i,t,1} = \tau_1, \cdots, T_{i,t,K} = \tau_K), \tag{13}$$

$$\mathbb{V}_{i,t}(\varrho, \tau) = \mathbb{P}(\hat{\mathbf{P}}_{i,[n>t]}, \hat{\mathbf{T}}_{i,[n>t]} | P_{i,t} = \varrho, T_{i,t,1} = \tau_1, \cdots, T_{i,t,K} = \tau_K) \quad (1 \le t \le n_i). \tag{14}$$

where

$$\tau = (\tau_1, \cdots, \tau_K), \hat{\mathbf{P}}_{i,[n\le t]} = (\hat{P}_{i,1}, \cdots, \hat{P}_{i,t}), \hat{\mathbf{P}}_{i,[n>t]} = (\hat{P}_{i,t+1}, \cdots, \hat{P}_{i,n_i}),$$
$$\hat{\mathbf{T}}_{i,[n\le t]} = (\hat{T}_{i,1}, \cdots, \hat{T}_{i,t}), \hat{\mathbf{T}}_{i,[n>t]} = (\hat{T}_{i,t+1}, \cdots, \hat{T}_{i,n_i}).$$

The forward and backward variables can be computed using the following dynamic programming iteration formula,

$$\mathbb{U}_{i,1}(\varrho, \tau) = \mathbb{P}(P_{i,1} = \varrho)\mathbb{P}(\hat{P}_{i,1}|P_{i,1} = \varrho) \prod_{k=1}^{K} \mathbb{P}(T_{i,1,k} = \tau_k)\mathbb{P}(\hat{T}_{i,1,k}|T_{i,1,k} = \tau_k), \tag{15}$$

$$\begin{aligned}\mathbb{U}_{i,t+1}(\varrho, \tau) = &\sum_{\varrho^*=1}^{L} \sum_{\tau_1^*=0}^{1} \cdots \sum_{\tau_K^*=0}^{1} \mathbb{U}_{i,t}(\varrho^*, \tau^*)\mathbb{P}(P_{i,t+1} = \varrho | P_{i,t} = \varrho^*)\mathbb{P}(\hat{P}_{i,t+1}|P_{i,t+1} = \varrho) \\ &\times \prod_{k=1}^{K} \mathbb{P}(T_{i,t+1,k} = \tau_k | T_{i,t,k} = \tau_k^*)\mathbb{P}(\hat{T}_{i,t+1,k}|T_{i,t+1,k} = \tau_k); \mathbb{V}_{i,n_i}(\varrho, \tau) = 1,\end{aligned} \tag{16}$$

$$\begin{aligned}\mathbb{V}_{i,t-1}(\varrho, \tau) = &\sum_{\varrho^*=1}^{L} \sum_{\tau_1^*=0}^{1} \cdots \sum_{\tau_K^*=0}^{1} \mathbb{V}_{i,t}(\varrho^*, \tau^*)\mathbb{P}(P_{i,t} = \varrho^* | P_{i,t-1} = \varrho)\mathbb{P}(\hat{P}_{i,t-1}|P_{i,t-1} = \varrho) \\ &\times \prod_{k=1}^{K} \mathbb{P}(T_{i,t-1,k} = \tau_k^* | T_{i,t,k} = \tau_k)\mathbb{P}(\hat{T}_{i,t-1,k}|T_{i,t-1,k} = \tau_k).\end{aligned} \tag{17}$$

Note that

$$\mathbb{P}(\hat{\mathbf{T}}, \hat{\mathbf{P}} | \boldsymbol{\theta}^{(t)}) = \sum_{\varrho \in \mathcal{P}} \sum_{\tau \in \mathcal{T}} \sum_{i=1}^{m} \mathbb{U}_{i,n_i}(\varrho, \tau). \tag{18}$$

The expectations can be computed using the following formulas,

$$\mathbb{E}\left[\mathbb{N}(P_{\cdot,1} = \varrho)|\boldsymbol{\theta}^{(t)}\right] = \sum_{i=1}^{m} \sum_{\tau \in \mathcal{T}} \mathbb{U}_{i,1}(\varrho, \tau)\mathbb{V}_{i,1}(\varrho, \tau) \Big/ \mathbb{P}(\hat{\mathbf{T}}, \hat{\mathbf{P}} | \boldsymbol{\theta}^{(t)}), \tag{19}$$

$$\begin{aligned}\mathbb{E}\left[\mathbb{N}(P_{\cdot,t-1} = \varrho, P_{\cdot,t} = \varrho^*)|\boldsymbol{\theta}^{(t)}\right] = &\sum_{i=1}^{m} \sum_{\tau \in \mathcal{T}} \sum_{\tau^* \in \mathcal{T}} \sum_{t=1}^{n_i-1} \mathbb{U}_{i,t}(\varrho, \tau)\mathbb{V}_{i,t+1}(\varrho^*, \tau^*)\mathbb{P}(P_{i,t+1} = \varrho^* | P_{i,t} = \varrho)\mathbb{P}(\hat{P}_{i,t+1}|P_{i,t+1} = \varrho^*) \\ &\times \prod_{k=1}^{K} \mathbb{P}(T_{i,t,k} = \tau_k | T_{i,t+1,k} = \tau_k^*)\mathbb{P}(\hat{T}_{i,t+1,k}|T_{i,t+1,k} = \tau_k^*) \Big/ \mathbb{P}(\hat{\mathbf{T}}, \hat{\mathbf{P}} | \boldsymbol{\theta}^{(t)}),\end{aligned} \tag{20}$$

$$\mathbb{E}\left[\mathbb{N}(P_{\cdot,t} = \varrho, \hat{P}_{\cdot,t} = \varrho^*)|\boldsymbol{\theta}^{(t)}\right] = \sum_{\tau \in \mathcal{T}} \sum_{\tau^* \in \mathcal{T}} \sum_{(\hat{P}_{i,t}, \hat{T}_{i,t}) = (\varrho^*, \tau^*)} \mathbb{U}_{i,t}(\varrho, \tau)\mathbb{V}_{i,t}(\varrho, \tau) \Big/ \mathbb{P}(\hat{\mathbf{T}}, \hat{\mathbf{P}} | \boldsymbol{\theta}^{(t)}), \tag{21}$$

$$\mathbb{E}\left[\mathbb{N}(T_{\cdot,1,k} = \tau)|\boldsymbol{\theta}^{(t)}\right] = \sum_{\varrho \in \mathcal{P}} \mathbb{U}_{i,1}(\varrho, \tau)\mathbb{V}_{i,1}(\varrho, \tau) \Big/ \mathbb{P}(\hat{\mathbf{T}}, \hat{\mathbf{P}} | \boldsymbol{\theta}^{(t)}), \tag{22}$$

$$\mathbb{E}\left[\mathbb{N}(T_{\cdot,t-1,k}=j,T_{\cdot,t,k}=\tau^*,P_{\cdot,t}=\varrho)|\boldsymbol{\theta}^{(t)}\right]=\sum_{\varrho^*\in\mathcal{P}}\sum_{t=1}^{n_i-1}\mathbb{U}_{i,t}(\varrho,\tau)\mathbb{V}_{i,t+1}(\varrho^*,\tau^*)\mathbb{P}(P_{i,t+1}=\varrho^*|P_{i,t}=\varrho)\mathbb{P}(\hat{P}_{i,t+1}|P_{i,t+1}=\varrho^*)$$
$$\times\prod_{k=1}^{K}\mathbb{P}(T_{i,t,k}=\tau_k|T_{i,t+1,k}=\tau_k^*)\mathbb{P}(\hat{T}_{i,t+1,k}|T_{i,t+1,k}=\tau_k^*)\Big/\mathbb{P}(\hat{\mathbf{T}},\hat{\mathbf{P}}|\boldsymbol{\theta}^{(t)}),\tag{23}$$

$$\mathbb{E}\left[\mathbb{N}(T_{\cdot,t,k}=\tau,\hat{T}_{\cdot,t,k}=\tau^*)|\boldsymbol{\theta}^{(t)}\right]=\sum_{\varrho\in\mathcal{P}}\sum_{\varrho^*\in\mathcal{P}}\sum_{(\hat{P}_{i,t},\hat{T}_{i,t})=(\varrho^*,\tau^*)}\mathbb{U}_{i,t}(\varrho,\tau)\mathbb{V}_{i,t}(\varrho,\tau)\Big/\mathbb{P}(\hat{\mathbf{T}},\hat{\mathbf{P}}|\boldsymbol{\theta}^{(t)}).\tag{24}$$

C. The Degenerated Cases

When only tool recognition is considered, we get the degenerated likelihood with parameters $\boldsymbol{\theta}=(\boldsymbol{\beta},\mathcal{A},\mathcal{B})$ as the degenerated parameters, resulting in the following Q-function,

$$\begin{aligned}Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)})=&\sum_{\tau\in\mathcal{T}}\sum_{j=0}^{1}((1-j)\log(1-\beta_\tau)+j\log(\beta_\tau))\mathbb{E}\left[\mathbb{N}(T_{\cdot,1,\tau}=j|\boldsymbol{\theta}^{(s)})\right]\\&+\sum_{\tau\in\mathcal{T}}\sum_{i=0}^{1}\sum_{j=0}^{1}\log\mathbf{A}_\tau(i,j)\mathbb{E}\left[\mathbb{N}(T_{\cdot,t-1,\tau}=i,T_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right]\\&+\sum_{\tau\in\mathcal{T}}\sum_{i=0}^{1}\sum_{j=0}^{1}\log\mathbf{B}_\tau(i,j)\mathbb{E}\left[\mathbb{N}(T_{\cdot,t,\tau}=i,\hat{T}_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right],\end{aligned}\tag{25}$$

where $\boldsymbol{\theta}^{(s)}$ is the parameter estimation in the s-th iteration of the EM algorithm, and

$$\begin{aligned}\mathbb{E}\left[\mathbb{N}(T_{\cdot,1,\tau}=j|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,1,\tau}=j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\\mathbb{E}\left[\mathbb{N}(T_{\cdot,t-1,\tau}=i,T_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,t-1,\tau}=i,T_{i,t,\tau}=j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\\mathbb{E}\left[\mathbb{N}(T_{\cdot,t,\tau}=i,\hat{T}_{\cdot,t,\tau}=j|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(T_{i,t,\tau}=i,\hat{T}_{i,t,\tau}=j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}).\end{aligned}$$

Similarly, when only phase recognition is considered, we have

$$\begin{aligned}Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(s)})=&\sum_{\varrho\in\mathcal{P}}\log\alpha_\varrho\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]\\&+\sum_{\varrho_i\in\mathcal{P}}\sum_{\varrho_j\in\mathcal{P}}\log\mathbf{A}(\varrho_i,\varrho_j)\mathbb{E}\left[\mathbb{N}(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]\\&+\sum_{\varrho_i\in\mathcal{P}}\sum_{\varrho_j\in\mathcal{P}}\log\mathbf{B}(\varrho_i,\varrho_j)\mathbb{E}\left[\mathbb{N}(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right],\end{aligned}\tag{26}$$

where

$$\begin{aligned}\mathbb{E}\left[\mathbb{N}(P_{\cdot,1}=\varrho|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,1}=\varrho)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\\mathbb{E}\left[\mathbb{N}(P_{\cdot,t-1}=\varrho_i,P_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,t-1}=\varrho_i,P_{i,t}=\varrho_j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}),\\\mathbb{E}\left[\mathbb{N}(P_{\cdot,t}=\varrho_i,\hat{P}_{\cdot,t}=\varrho_j|\boldsymbol{\theta}^{(s)})\right]&=\sum_{i=1}^{m}\sum_{\mathcal{I}_i}\mathbb{I}(P_{i,t}=\varrho_i,\hat{P}_{i,t}=\varrho_j)\mathbb{P}(\mathcal{I}_i|\mathcal{I}_i^{obs},\boldsymbol{\theta}^{(s)}).\end{aligned}$$

By following the same procedure outlined in Section II-A, we can derive the iteration formula of the EM algorithm for the degenerate case. The E-step in the degenerate model can be computed quickly using a similar forward-backward procedure as described in Section II-B.

III. Measurements for Performance Evaluation

Following [1], [2] and [3], we choose *average precision* (AP) and *mean average precision* (mAP) as the primary performance measurement for tool recognition. Let $\mathcal{F}$ be the collection of m key frames in the testing videos, $T_{f,\tau}$ be the true presence label of tool τ, $\pi_{f,\tau}$ be the predictive probability of tool τ to appear in a key frame $f \in \mathcal{F}$ output by a surgical tool recognizer $\mathcal{M}$. For a given cutoff parameter $\lambda \in (0,1)$, the precision and recall of recognizer $\mathcal{M}$ for recognizing tool τ under cutoff λ are defined as:

$$P_\tau(\lambda) = \frac{\sum_{f\in\mathcal{F}} \mathbb{I}(T_{f,\tau} = \mathbb{I}(\pi_{f,\tau} > \lambda) = 1)}{\sum_{f\in\mathcal{F}} \mathbb{I}(\pi_{f,\tau} > \lambda)},$$
$$R_\tau(\lambda) = \frac{\sum_{f\in\mathcal{F}} \mathbb{I}(T_{f,\tau} = \mathbb{I}(\pi_{f,\tau} > \lambda) = 1)}{\sum_{f\in\mathcal{F}} \mathbb{I}(T_{f,\tau} = 1)}.$$

For any tool $\tau \in \mathcal{T}$, the AP of recognizing τ by recognizer $\mathcal{M}$, which is defined as the area under the corresponding precision-recall curve, can be calculated as follows:

$$\text{AP}_\tau = \sum_{i=1}^{m} P_\tau(\lambda_{i-1,\tau})(R_\tau(\lambda_{i,\tau}) - R_\tau(\lambda_{i-1,\tau})), \tag{27}$$

where $\{\lambda_{i,\tau}\}_{1\le i\le m}$ are the ordered statistics of $\{\pi_{f,\tau}\}_{f\in\mathcal{F}}$ with $\lambda_{0,\tau} = 0$. To evaluate the overall performance of recognizer $\mathcal{M}$, we averaged the AP_τ's of K tools to form mAP:

$$\text{mAP} = \frac{1}{K}\sum_{\tau\in\mathcal{T}} \text{AP}_\tau. \tag{28}$$

Following [3], we selected the *F1-score* defined below as the primary metric for performance evaluation of phase recognition. Let $\mathcal{F}$ denote the set of m key frames in the testing videos, P_f and $\hat{P}_f$ represent the true label and predicted labels of phase about a key frame $f \in \mathcal{F}$. The precision and recall of a surgical phase classifier $\mathcal{M}$ for classifying any phase $\varrho \in \mathcal{P}$ are defined as follows:

$$P_\varrho = \frac{\sum_{f\in\mathcal{F}} \mathbb{I}(P_f = \hat{P}_f = \varrho)}{\sum_{f\in\mathcal{F}} \mathbb{I}(\hat{P}_f = \varrho)},$$
$$R_\varrho = \frac{\sum_{f\in\mathcal{F}} \mathbb{I}(P_f = \hat{P}_f = \varrho)}{\sum_{f\in\mathcal{F}} \mathbb{I}(P_f = \varrho)}.$$

For any phase $\varrho \in \mathcal{P}$, the F1-score of classifying ϱ by recognizer $\mathcal{M}$ is defined as:

$$\text{F1}_\varrho = \frac{2 \cdot P_\varrho \cdot R_\varrho}{P_\varrho + R_\varrho}. \tag{29}$$

To calculate the overall performance of recognizer $\mathcal{M}$ on all surgical phases, we averaged the F1-score$_\varrho$'s of L phases as below:

$$\text{mF1} = \frac{1}{L}\sum_{\tau\in\mathcal{T}} \text{F1}_\varrho. \tag{30}$$

References

[1] A. P. Twinanda, S. Shehata, D. Mutter, J. Marescaux, M. de Mathelin, and N. Padoy, "Endonet: a deep architecture for recognition tasks on laparoscopic videos," *IEEE Transactions on Medical Imaging*, vol. 36, no. 1, pp. 86–97, 2017.

[2] A. Jin, S. Yeung, J. Jopling, J. Krause, D. Azagury, A. Milstein, and L. Fei-Fei, "Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018, pp. 691–699.

[3] Y. Jin, H. Li, Q. Dou, H. Chen, J. Qin, C. Fu, and P. Heng, "Multi-task recurrent convolutional network with correlation loss for surgical video analysis," *Medical Image Analysis*, vol. 59, no. 1, pp. 1–14, 2019.

[4] S. Kondo, "Lapformer: surgical tool detection in laparoscopic surgical video using transformer architecture," *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, vol. 9, no. 3, pp. 302–307, 2021.

[5] R. Tao, X. Zou, and G. Zheng, "Last: Latent space-constrained transformers for automatic surgical phase recognition and tool presence detection," *IEEE Transactions on Medical Imaging*, vol. 42, no. 11, pp. 3256–3268, 2023.

[6] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 39, no. 1, pp. 1–22, 1977.

[7] L. E. Baum, T. Petrie, G. W. Soules, and N. Weiss, "A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains," *Annals of Mathematical Statistics*, vol. 41, no. 1, pp. 164–171, 1970.