SPATIAL-TEMPORAL ATTENTION FOR VIDEO-BASED ASSESSMENT OF INTRAOPERATIVE SURGICAL SKILL

by
Bohua Wan

A thesis submitted to The Johns Hopkins University in conformity with the requirements for the degree of Master of Science in Engineering

Baltimore, Maryland
May, 2022

Abstract

Surgical skill assessment using videos from operating room is a crucial aspect in surgeon evaluations. In this thesis, we first present an analysis of attention mechanism for video-based assessment of intraoperative surgical skills. We propose a novel method that uses spatio-temporal attention, where the spatial attention module is supervised by instrument tip trajectories. It is now unequivocal that instrument tip trajectories are most informative of surgical skill. We hypothesize that supervising attention with instrument motion will improve performance by regularizing the network to use the most relevant information. We compare our method against a network in our previous work that uses unsupervised spatio-temporal attention and a network using a multi-task learning framework as baseline. Our ablation studies show supervising the spatial attention help improve model's performance and generalizability in internal and external validations.

Next, we present a two-stage approach to extract temporal attention for the whole video instead of short clips. Our approach first extract features of each frame of a given video. Then, we train a simple temporal network with temporal attention on top of the extracted features. With the trained temporal network, we can extract temporal attention that is normalized across the video.

Finally, we describe three semi-supervised domain adaptation (SSDA) methods for improving our models' performance on external validation. We explored the Vanilla SSDA method that simply include target domain samples to source domain. The next method is the Group Distributionally Robust Supervised Learning (Group-DRSL) method, which groups data according to their domain. This method assign weights to samples according to the group and adversarially optimize the weights. The third approach adds class weighting to the second approach. Experiments result show Group-DRSL performs the best and consistently improves models' performance in external validations.

Thesis Reader

Dr. Gregory D. Hager (Primary Advisor)
Mandell Bellmore Professor
Department of Computer Science
Johns Hopkins University

This thesis is dedicated to my family.

For their endless love and support.

Acknowledgements

This thesis would not have been possible without the guidance and help of many people who in one way or another contributed. First of all, I would like to express my greatest gratitude to Dr. S. Swaroop Vedula for giving me the opportunity to pursue research in a talented, warm, and welcoming team and supporting me in research and study. I am extremely grateful for his continuous guidance and encouragement throughout this project. It was a true pleasure working under his supervision. I am thankful to him for always being available for discussion and suggestions. Numerous ideas sparked from our discussions.

I am also more than grateful to Dr. Gregory D. Hager for being my advisor on this project and providing insightful comments on the work. His thoughtful questions taught me the importance of scientific thinking and pointed the direction of my research. I am thankful to receive his guidance and support.

I would also like to extend deepest my gratitude to Dr. Shameema Sikder for providing insightful opinions and feedback from a surgeon's perspective. I would also like to thank Dr. Vishal Patel for his valuable comments and discussions. I am also grateful to Dr. Anqi Liu for her helpful advice and guidance.

I want to thank Michael Peven for providing generous help to my work. I am very grateful for his effort in editing our previous publication and for providing insightful feedback on my project. His suggestions on my experiments are greatly helpful.

Finally, I am truly grateful to my family for their unconditional love and constant

support both mentally and financially.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Surgeries are being performed hundreds of millions of times across the world each year [1]. Improving the proficiency of the surgeons is therefore of paramount importance to improve patient safety and outcome [2]. Surgical skill assessment is a crucial component in programs improving surgeons' proficiency and reducing clinical errors [3]. Conventional surgical skill assessment methods include questionnaires or form based-methods (Objective Assessment of Skills in Intraocular Surgery [4], Global Rating Assessment of Skills in Intraocular Surgery [5], Human Reliability Analysis of Cataract Surgery [6], OSCAR [7], Objective Structured Assessment of Cataract [8], and Objective Structured Assessment of Technical Skill [9]), and crowdsourcing based methods [10, 11]. These methods are either subjective and costly due to the involvement of experts or too generic to provide a procedure-specific assessment [12].

Surgical data science approaches allow data-driven, objective, and automated video-based assessment (VBA) of intraoperative surgical skills [13–16]. These approaches can be categorized into three categories: 1. conventional computer vision methods to detect and segment structures in video images [12]; 2. networks to extract instrument motion and methods to compute features describing it [17–21]; and 3. networks for end-to-end analysis of video images to regress on surgical skill [22–25]. Our work presented in this thesis falls in the third category.

The purpose of our work is to develop and validate a novel method for VBA of intraoperative surgical skill algorithm. We analyze the effect of spatial-temporal attention for VBA of intraoperative surgical skills. Our experiments are done in a data-scarce setting. The amount of available data is limited, which will cause the models to be more prone to over-fitting. We also explored ways to improve a model's generalizability in terms of its performance in external validation under covariate shift and label shift The purpose of our work is to develop and validate a novel method for VBA of intraoperative surgical skill algorithm.

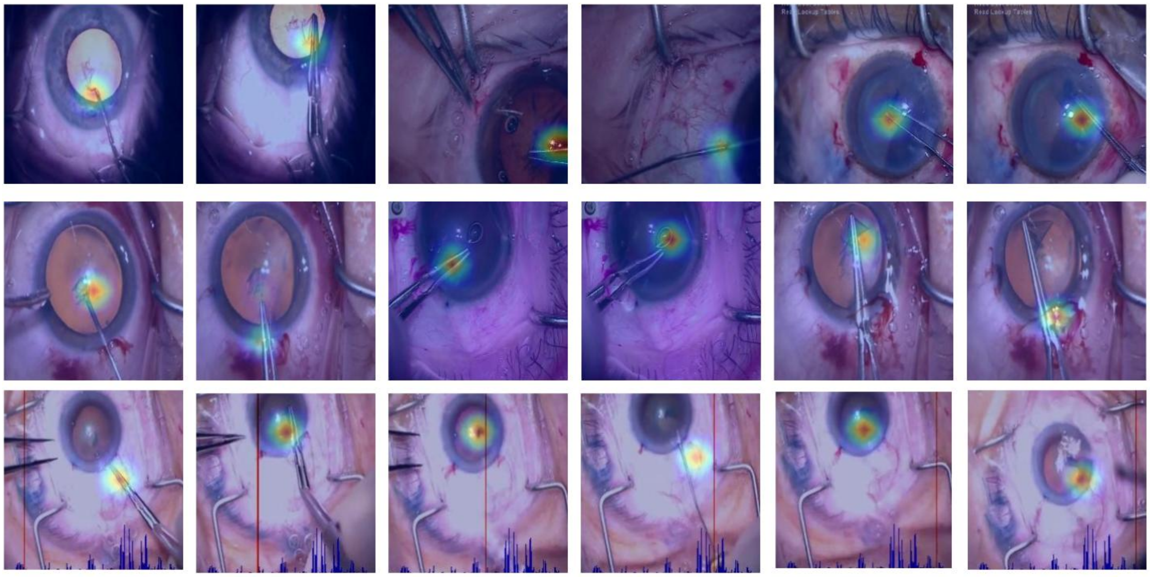

Figure 1-1. An illustration of spatial attention and temporal attention. The top two rows show the generated spatial attention for test samples. The coloring of the attention follows the Jet colormap. The red and bright parts are where attention values are high. The bottom row shows both spatial and temporal attention. The heights of the blue bars at the bottom of the images show how important the corresponding frame is. The red bar shows the progress of the video and locates the current frame.

Attention mechanism has been used to improve model performance in various areas [26] such as image classification [27, 28], object detection [29, 30], and action recognition [31, 32]. [25] and [33] have also shown using spatial attention that guides the model on where to focus in each frame helps improve VBA of surgical skill. However, their approaches are purely unsupervised. The learned attention map may focus on spurious correlations and cause the model to over-fit the training data. These problems are particularly evident when the amount of training data is scarce. To solve these problems, we propose to supervise the spatial attention such that the learned attention will focus on regions where most skill-relevant information is shown using signals such as instrument trajectories. Several studies have shown that what surgeons do with their instruments contains critical information for assessing their skill [18, 34, 35]. Furthermore, in microscopic surgical procedures, the tissue is minimally manipulated and thus, instrument motion contains most information for assessing the skill. Therefore, we hypothesize that supervising the spatial attention with instrument trajectories would improve the model's performance and generalizability since the model would focus only on regions where most information is contained and ignore spurious correlations. Our ablation study shows supervising the attention indeed helps the model be more generalizable compared with the unsupervised attention model and multi-task learning.

Temporal attention has also been shown to improve the model's performance [25, 36]. However, due to computation limitations, temporal attention is only normalized and used within batches in most existing works. In this work, we overcome this problem by using a two-step approach. Our approach first extracts features using a pre-trained network and then builds another temporal network on top of the features. Using our method, temporal attention for the whole video can be computed. This will allow surgeons and other users to examine which part of the video is considered important by the model.

External validation is crucial for evaluating a model's utility. Existing methods [22, 25, 37] only evaluate their methods internally, which means the test samples are drawn from the same dataset and the same distribution. However, in real application scenarios, the test samples rarely come from the same distribution as the development dataset. Therefore, evaluating the model with external data or data from another distribution is necessary to assess the applicability of the model. Our external validation experiments show that models with or without attention perform significantly worse tested with external data, further details are discussed in section 4.5. To improve models' performance in external validation, we explored several semi-supervised domain adaptations (SSDA) and distributionally robust supervised learning (DRSL) techniques. Although we were able to improve the model's performance, there is still an evident gap between the model's performance in internal validation and external validation.

Our contributions can be summarized as:

1. improving spatial attention module by supervising it with instrument tip locations, which represent the inherent structure in the input images that have been shown to be highly informative for the prediction task (an illustration of the learned attention maps is shown in Fig. 1-1);
2. comparing different ways to attend to the image features in each frame; and
3. evaluating a multi-task learning framework using instrument trajectories as an ancillary task.
4. developing a two-stage approach to generate temporal attention for whole videos of length more than 20 minutes.
5. evaluating the effectiveness of several SSDA and DRSL approaches in improving VBA models on our dataset.

Chapter 2

Related Work

2.1 Video-based assessment of surgical skills

VBA of surgical skills approaches can be split into three categories as mentioned in chapter 1.

In the first category, in [12], Baghdadi, et.al., extract features from edge and line detections, and object detection with geometric features. These features are then used to calculate metrics that were used to classify surgical skills for pelvic lymph node dissection during radical cystectomy.

In the second category, in [17], Law, et.al., train a network to track points of articulations in the instruments in a surgical robot as keypoints. They compute motion features for individual keypoints, correlations between keypoints, and smoothness of motion at the keypoints as features. These features are input to an SVM classifier to predict the skill level. Lin, et.al., use geometric methods and optical flow to track instrument tips, which are used to compute motion metrics and to compare them among surgeons with different levels of experience [18]. In [19], surgical skill score is proxied by the clearness of the operating field (COF). COF is estimated from color features extracted from video images. COF features concatenated with image features extracted using ResNet-101 are then analyzed using multi-layer perceptrons. The score and weights of each frame are then computed. A weighted average of the scores is calculated as the skill score for the video. In [20], Lavanchy, et.al., detect bounding boxes around surgical instruments in video images with a neural network. They then compute motion descriptors that are input to a regression model for skill score prediction. Similarly, instruments are localized by [21] using a region proposal network, which is then used to analyze tool usage patterns, range of movements, the economy of motion, etc.

In the third category, in [22], Liu, et.al., designed a 4-pathway framework that analyzes visual, event, tool, and COF inputs. Path independent modules in this framework calculate skill score sequences for each path. Path-dependent modules calculate weight sequences. A final weighted skill score is then computed from the two types of sequences. In [23], Kitaguchi, et.al., proposed a Inception-v1 I3D based 2-stream model [24] that predicts the skill level. Hira et.al. [25], devised a framework that uses convolutional neural networks to extract frame-level image features and then uses the long short-term memory (LSTM) network to extract temporal features of the video. They augment this framework with unsupervised spatial and temporal attention modules. They generate spatial attention by combining spatial and temporal features by directly adding the hidden state from LSTM to image features.

Our work is built on top of [25]'s work. We extend the analysis of spatial-temporal attention and propose to supervise spatial attention. We further study the generalizability of spatial-temporal attention models.

2.2 Attention mechanism

Attention modules are first introduced for machine translation [38]. It then became extremely popular within the artificial intelligence community as an essential building block of neural architectures [39]. A large number of applications of attention can be found in neural language processing [40], speech [41], and computer vision [42].

The intuition behind the attention mechanism is to focus on the most related parts of the input. Attention modules can be categorized into soft/global, hard, and local attention from the positioning perspective [39]. Soft/global attention introduced in [38] uses a weighted average of all hidden states of the input to build a features vector. Hard attention is introduced in [43] stochastically sampling the hidden states of the input. Computation efficiency is one of the advantages of using hard attention. Local attention [44] is in between the soft/global attention and the hard attention. It calculates a weighted average of hidden states within a window. Our analysis in this work focuses on soft/global attention.

Attention can also be split into distinctive, co-attention, and self-attention [38]. Distinctive attention calculates the attention from two distinct input sources. Co-attention is computed from multiple input sources. Self-attention is computed from a single input source. Transformers [45, 46] are family of neural networks that adopt self-attention. The attention mechanism discussed in this thesis falls in the self-attention category according to [38]'s categorization. However, the attention mechanism in this paper is different from the attention used in transformers. In this thesis, attention is computed per pixel and no between-pixels or between patches modeling is involved. The attention value of each pixel is computed from the feature representation of that pixel alone.

2.2.1 Supervised attention

Most of the previous works supervise the attention jointly with task loss (cross-entropy loss, hinge loss, and others). However, jointly supervising the attention with task loss may cause the learned attention to focus on irrelevant parts of the input and cause over-fitting [47]. Furthermore, the attention map learned with task loss does not correlate with human attention [47]. [48] shows that using segmentation question-answering link to supervise the attention helps improve model performance. They use pre-collected semantic segmentation results to supervise the attention. The collected semantic segmentation results denote which parts of the image is related to the question-answering task. [49] shows that supervising the attention serves as a form of regularization. Our work shows similar results that using supervised attention help improve model performance and generalizability.

2.3 Domain adaptation

Domain adaptation is a line of research that studies how to improve models' performance on the target domain when the models are trained on the source domain. The term "domain" is interchangeable with the "distribution". For ease of writing, we also refer to the dataset as domain. Most existing domain adaptation research focuses on the setting where the source domain contains abundant data [50–56]. They typically try to align the feature distributions and prediction distributions between the source and target domain. [57, 58] have focused on the setting where the amount of labels in the source domain is also limited. Their work in few-shot settings could be beneficial to VBA research.

2.3.1 Video-based domain adaptation

Video-based domain adaptation has received much less attention compared to image-based domain adaptation research [56, 59, 60]. [61] studied how to transfer knowledge from image-based face recognition to video-based face recognition. [51, 62, 63] tried to solve the video-based domain adaptation problem. [62] proposed two video domain adaptation datasets and a temporal attentive adversarial adaptation network that focuses on aligning the temporal dynamics between source and target domain. [63] proposed two approaches: 1. model clips of videos as points in latent space and then successively learning adaptive kernels between points in the source domain and those in the target domain; 2. a typical adversarial domain adaptation approach. [51]

focused on open-set problems in video-based domain adaptation.

2.3.2 Semi-supervised domain adaptation

Semi-supervised domain adaptation uses a small number of labeled target samples to help adapt the shift between the two domains. Obtaining a small number of labeled target samples is typically feasible in VBA tasks, therefore it is suitable in my setting. [52] tried to learn invariant representation and risks to solve the domain adaptation problem. [53] proposed a few-shot min-max entropy approach under the adversarial optimization framework. [50] proposed a typical two-stream model and used group labels that group source samples labels and target samples label to allow the model to better learn the transferable knowledge. DRSL is also related to SSDA. The goal of DRSL is to train a distributionally robust model whose performance does not drop significantly when distribution shifts occur in the external validation dataset.

Our work in this thesis follows SSDA domain adaptation settings, where we use 10 target samples to improve model performance in external validation. We explore a Group-DRSL-based approach and the results show an evident improvement over baseline models.

Chapter 3

Methodology

We first introduce the overall structure of our model. Next, we show how instrument trajectories are used to supervise spatial attention such that only the most relevant and informative features around the instrument tips are used by the model to make the final prediction. We then explain two variants of the spatial attention module used in this study, and a baseline multi-task learning model. Next, we introduce the temporal attention module used in our model. We then show how to compute temporal attention for long videos under computational limits. Finally, we introduce the semi-supervised domain adaptation techniques we used to improve the model performance in external validation. The materials presented in section 3.1, 3.2, and 3.3 are reproduced from our previous work [64].

3.1 Overall structure

We formulate skill assessment as binary classification (expert/novice) using an RGB video as input. The overall architecture of our network, based on [25] is shown in Fig. 3-1. Given all frames of the input video, a CNN backbone extracts per-frame image features. These features are passed into the spatial attention module, which produces per-frame features. These features are concatenated together as input to the temporal attention module; the final hidden state of the temporal model is passed into a linear

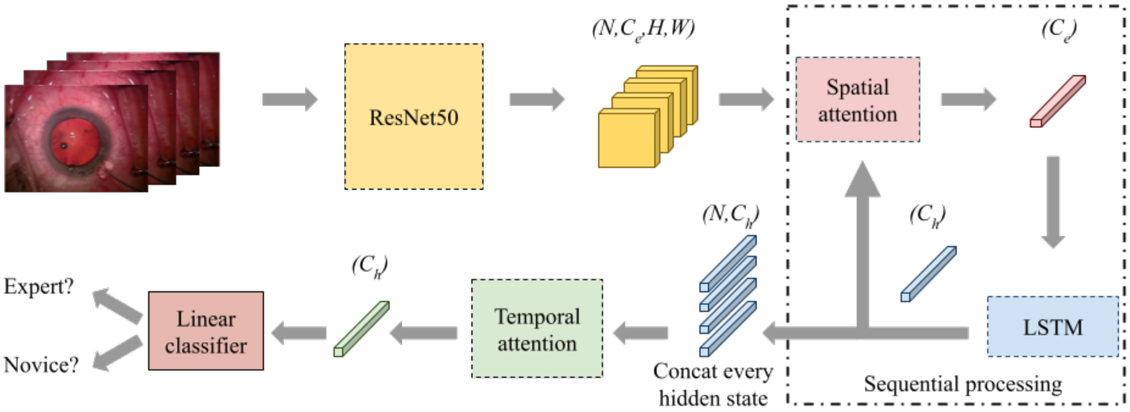

Figure 3-1. An overview of our model. The "Spatial attention" box is further illustrated in Fig. 3-2 and discussed in section 3.2.1. The "Temporal attention" box is discussed in section 3.3. The C_e and C_h are the feature dimensions of the image features and hidden state, respectively.

classifier to produce a skill label.

3.2 Supervised spatial attention

3.2.1 Spatial attention module

In this subsection, we introduce how the spatial attention map is calculated. We explain the supervision mechanism for the spatial attention module and discuss two ways to attend on the image features using the spatial attention map - aggregation and selection. Finally, we explain multi-task learning of keypoint localization as an implicit way to supervise the spatial attention map.

As shown in Fig. 3-2, the spatial attention module takes two inputs. First, pixel-wise image features (appearance features) $p_{i,m,n}$ from pixels $m \in [1,H], n \in [1,W]$, where H, W are the height and width of the CNN backbone encoded image features, and frame $i \in [1,N]$. The second input is the LSTM hidden state (spatio-temporal features) $\mathbf{h}_i$ at that frame. The score for overall attention map $\mathbf{A}_i^{spatial}$ at each pixel is calculated as follows:

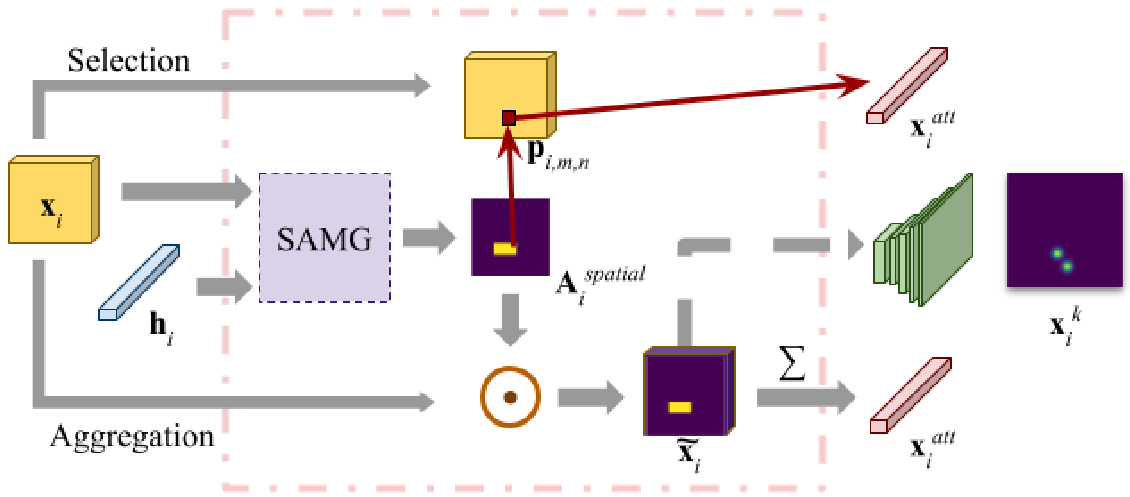

Figure 3-2. The spatial attention module. The upper and lower streams correspond to the selection and aggregation scheme, respectively. In practice, we use one scheme and not both. The pink dashed box outlines the spatial attention module. The dashed arrow shows the pathway for the multi-task learning model used for comparison. The SAMG box denotes the process to compute the spatial attention map, $\odot$ is a dot product, and $\sum$ is a summation along the height and width dimensions. The green stacked cubicles following the dashed arrow represent five layers of transposed convolutional layers.

$$f_{\text{appearance}} = \mathbf{M}_a \mathbf{p}_{i,m,n} \tag{3.1}$$

$$f_{\text{spatio-temporal}} = \mathbf{M}_s \mathbf{h}_i \tag{3.2}$$

$$f_{m,n}^{\text{overall}} = \mathbf{M}_o \sigma(f_{\text{appearance}} + f_{\text{spatio-temporal}}) \tag{3.3}$$

$$att_{m,n}^{spatial} = \frac{\exp(f_{m,n}^{\text{overall}})}{\sum_{m,n} \exp(f_{m,n}^{\text{overall}})} \tag{3.4}$$

where σ is the ReLU activation function. To ensure compatibility for the operations, $\mathbf{M}_a$, $\mathbf{M}_s$, and $\mathbf{M}_o$ are learned weight matrices for the appearance, spatio-temporal, and overall feature maps, respectively.

3.2.2 Supervised spatial attention map

Conventional attention models [33, 65], including the baseline model, learn attention maps with task-oriented loss (e.g. cross-entropy loss). These attention maps represent a layer of re-weighting or "attending to" the image features. However, without explicit supervision, they are not guaranteed to localize relevant regions in the images. Furthermore, without a large amount of training data, attention mechanisms could assign higher weights to regions having spurious correlations with the target label [47, 48]. Therefore, we hypothesize that explicitly supervising the attention map using specific information in the images can improve the accuracy of the model predictions.

In this work, we propose a method for explicit supervision of the attention map using instrument tip trajectories. Previous work has shown that instrument tip trajectories are highly informative of surgical skill [13, 66]. Specifically, we construct binary trajectory heat maps $\mathbf{B}_i$ for each frame i, combining the locations $s_{k,m,n}$ of all instrument tips, where s is a binary indicator variable denoting if instrument tip k is located at pixel coordinates m, n:

$$\forall b_{m,n} \in \mathbf{B}_i, \quad b_{m,n} = \begin{cases} 1 & \mathit{if} \sum\limits_k s_{k,m,n} \geq 1 \\ 0 & \mathit{otherwise} \end{cases} \tag{3.5}$$

For training, the overall loss function combines binary cross-entropy for skill classification L_{BCE} and the Dice coefficient between the spatial attention map $\mathbf{A}^{spatial}$ and the tool-tip heat map $\mathbf{B}$:

$$L_{Dice} = DL(\{\mathbf{A}_i^{spatial} | i \in [1, N]\}, \{\mathbf{B}_i | i \in [1, N]\}) \tag{3.6}$$

$$L = L_{BCE} + \lambda \cdot L_{Dice} \tag{3.7}$$

The weighting factor λ is empirically set to 0.5.

3.2.3 Aggregation

The first way of attending on the image features follows the aggregation scheme, where the attention-weighted image features $\widetilde{x}_i$ are summed over all pixels in each frame i:

$$\mathbf{x}_i^{att} = \sum \widetilde{\mathbf{x}}_i = \sum \mathbf{A}_i^{spatial} \cdot \mathbf{x}_i \tag{3.8}$$

The attention map $\mathbf{A}_i^{spatial}$ is supervised using the trajectory heat map, so the attended image feature vector should put higher weight on features around the instrument tip keypoints.

3.2.4 Selection

The second way of attending on the image features follows the selection scheme. The attended image feature vector $\mathbf{x}_i^{att}$ is computed as:

$$\hat{m}, \hat{n} = \arg\max_{m,n}(\{a_{m,n}^{spatial} | a_{m,n}^{spatial} \in \mathbf{A}_i^{spatial}\}), \tag{3.9}$$

$$\mathbf{x}_i^{att} = \mathbf{p}_{\hat{m},\hat{n}} \tag{3.10}$$

The selection scheme yields a single pixel from the image feature map, leading to more localized features than the aggregation scheme. This approach is similar to masking the image using a detected bounding box, but it avoids the need for a separate detection network.

3.2.5 Multi-task learning

Multi-task learning, which is an intuitive baseline model to compare a model with supervised attention maps, involves adding a keypoint localization branch to the network (Fig. 3-2) and an additional loss term. This branch is implemented using 5 layers of transposed convolutional layers to up-sample the attention weighted image features $\widetilde{\mathbf{x}_i}$ to output a predicted keypoint heat map. We use the method in [67] to compute ground truth Gaussian heat maps using keypoint annotations and mean squared error loss during training.

3.3 Temporal attention

The last part of the model is a temporal attention mechanism as can be seen from Fig. 3-1. The temporal attention $\mathbf{A}^{temp}$ is calculated from the hidden states of the LSTM, $\mathbf{H} = \{\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_N\}$ as follows:

$$\boldsymbol{\beta} = \mathbf{H}\mathbf{h}_N \tag{3.11}$$

$$\forall i \in [1, N], \quad \boldsymbol{A}_i^{temp} = \frac{exp(\boldsymbol{\beta}_i)}{\sum_{i \in [1,N]} \boldsymbol{\beta}_i} \tag{3.12}$$

$$\mathbf{h}_f = (\mathbf{A}^{temp})^T \mathbf{H} \tag{3.13}$$

The attended final hidden state, $\mathbf{h}_f$, is input to a linear layer for classification.

3.4 Temporal attention over the whole video

3.4.1 Background

In this section, we introduce how to generate temporal attention for the whole video instead of a short clip. Temporal attention not only helps improve model performance [32] but also shows the contribution of different parts of the video to the decision of the model. It would help surgeons to better understand the decision process of the model, and thus improve the explainability of the model. However, existing methods [25, 32] only generate temporal attention on a short clip of the video, due to computational limitations.

The lengths of intraoperative surgery videos could be as long as 30 minutes. It is simply computationally impossible to load a 30-minute-long video to RAM for processing. Therefore, the widely adopted approach is to sample a short clip of the video [68]. Then, only the short clip of the video is fed into the model for skill assessment and the temporal attention is only generated for that short clip of the video. Furthermore, although there are high redundancies between consecutive frames, the sampling strategies used by previous work [25] would inevitably drop a significant portion of the video, which would cause a loss of information that couldn't be neglected. [69] proposed an end-to-end approach that uses all frames of the video to solve this problem. The approach decreases the amount of memory needed for back-propagation using an approximation method. However, their approach still couldn't process a 30-minute-long video as their approach still requires loading all the images of the video to the memory at the beginning of their algorithm.

3.4.2 Two-stage approach

We propose a two-stage approach that first extracts the features of each frame in a video and then trains a simple temporal network on top of the extracted features. The temporal attention generated from the temporal network is then normalized across the whole video.

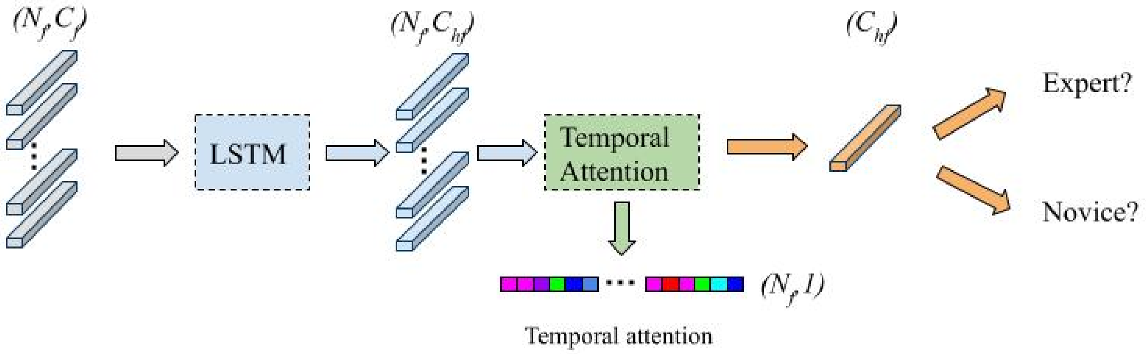

Figure 3-3. The temporal network on top of the extracted features. The temporal attention module is identical to that in figure 3-1 and is discussed in section 3.3. N_f denotes the total number of frames in a video. C_f denotes the length of the extracted feature vector of each frame.

The first step of our approach is to extract features of each frame from a given video. Figure 3-4 illustrates this process. Given a video, we use a sliding window to process all the frames. The window has a size of N. The N frames in the window are fed into our supervised attention model with the aggregation scheme, discussed in section 3.2.3. We opt for using the aggregation scheme because it shows better performance in external validation experiments. For each frame, we extracted both the hidden state from the LSTM of each frame as the temporal feature and the attended image feature as the spatial feature of each frame.

In the second stage, we train a temporal network on top of the extracted features. Figure 3-3 shows the structure of the temporal network. We use the same temporal attention module as the one discussed in section 3.3. This temporal network is supervised with skill labels. Using features of all frames in a video, we are able to generate temporal attention for the whole video using the temporal attention module. The extracted features are rich in information as they were extracted from the pretrained supervised attention model. Therefore, using this simple temporal network should be sufficient for generating the temporal attention and classifying the skill level.

3.5 Semi-supervised domain adaptation

Semi-supervised domain adaptation uses a small number of labeled samples from the target domain to help adapt the shift between the source domain and the target domain. In our specific case, we have two datasets, sampled from the two domains, or distributions. The videos from the two datasets are distributed significantly differently. Our objective is to improve our model's performance in the target domain.

We explored three semi-supervised domain approaches. The first approach referred to as Vanilla SSDA simply includes a few samples from the target domain to the source domain. The second approach is based on the Group Distributionally Robust Supervised Learning (Group-DRSL) [70]. We refer to it as the Group-DRSL approach in the following chapters. The third approach, Weighted-Group-DRSL, adopts class weighting to the second approach.

3.5.1 Problem setup

Given a video composed of a sequence of frames $X = \{x_i \in \mathbb{R}^{C \times H \times W} | i \in [0, F]\}$. Let y be the binary label denoting if the skill shown in the video is expert level or novice level. Let $D_{source} = \{(X, y)_i | i \in [0, N_s]\}$, $\hat{D}_{target} = \{(X, y)_i | i \in [0, N_{tl}]\}$, and $D_{target} = \{(X, y)_i | i \in [0, N_t]\}$ be the source domain, labeled target domain, and the target domain. $N_t >> N_{tl}$. We would like to obtain a robust model trained with samples from D_{source} and $\hat{D}_{target}$, such that it adapts to D_{target}.

3.5.2 Vanilla SSDA

The first approach is simple and straightforward. It will be used as the baseline model for comparison.

This method simply trains the model with all samples in the source domain together with those labeled target samples. Training together with the target samples may help the model better learn the transferable knowledge. Thus, the model's performance in the target domain should be improved.

3.5.3 Group-DRSL

The second approach extends the first approach by introducing the Group DRSL framework [70]. [70] propose to add structural assumption over normal DRSL. The assumption they adopted is the latent prior probability change assumption. Let $z \in S = \{1, ..., Z\}$ be a latent variable, it splits the dataset into Z groups. Let P denote the probability distribution of the source domain samples and Q denote that of the target domain samples. The assumption requires $P(X, y|z)$ to be the same between different datasets or domains. In our case, this assumption holds if the sampled target data are sampled independently and identically from the target distribution. Then, $P(X, y|z = target) = Q(X, y|z = target)$. Finally, we will use the adversarial training scheme, by adding a learnable weight $w(z)$ to reweight the loss for samples in the source domain and those in the target domain. Specifically, this approach tries to minimize the risk:

$$\min_{\theta} \sup_{w \in \widehat{W}} \frac{1}{N} \sum_{z=source}^{\{target,source\}} n_z w(z) L(z, \theta)$$
$$\widehat{W} = \{w \in \mathbb{R}^2 | \frac{1}{N} \sum_{z=source}^{\{target,source\}} n_z w(z) = 1, w \geq 0\}$$

where $L(z; \theta)$ is the averaged loss within-group z.

In practice, we will use one learnable parameter to simulate the $w(z = target)$. The $w(z = source)$ can be derived from $w(z = target)$:

$$w(z = source) = \frac{N - n_{target} w(z = target)}{n_{source}} \tag{3.14}$$

The learnable parameter tries to maximize the loss term and is the adversarial part of the training. It is frozen periodically following the typical adversarial training framework [71].

3.5.4 Weighted-Group-DRSL

In our case, the class labels in the source domain are well balanced. The numbers of positive and negative samples are roughly the same. However, in the target domain, the class labels are highly biased. There are very few positive samples in the target domain. Label shift is evident between the two domains. The problem with a biased dataset is that those samples from the minority class will have small contributions and influences to the learning simply because they appear less frequently compared to samples from other classes. Weighting samples is a typical approach to solving the imbalanced class label distribution problems and label shifts. Assigning a high weight to samples from minority classes will improve their contributions and influences during back-propagation as the assigned weight will magnify the gradient.

In practice, we assign weights for different samples according to their class label. Let $w_c(positive)$ be the weight assigned to the positive samples, and $w_c(negative)$ be the weight assigned to the negative samples. Then the loss term is rewrote as:

$$L = \frac{1}{N} \sum_{(X,y) \in D_{source} \bigcap \hat{D}_{target}} (\hat{y} w_c(positive) + (1 - \hat{y}) w_c(negative)) L(X, y) \tag{3.15}$$

$$\hat{y} = \begin{cases} 1 & \textit{if } y = positive \\ 0 & otherwise \end{cases}$$

Using this loss term with the Group-DRSL framework discussed in section 3.5.3 should yield better performance.

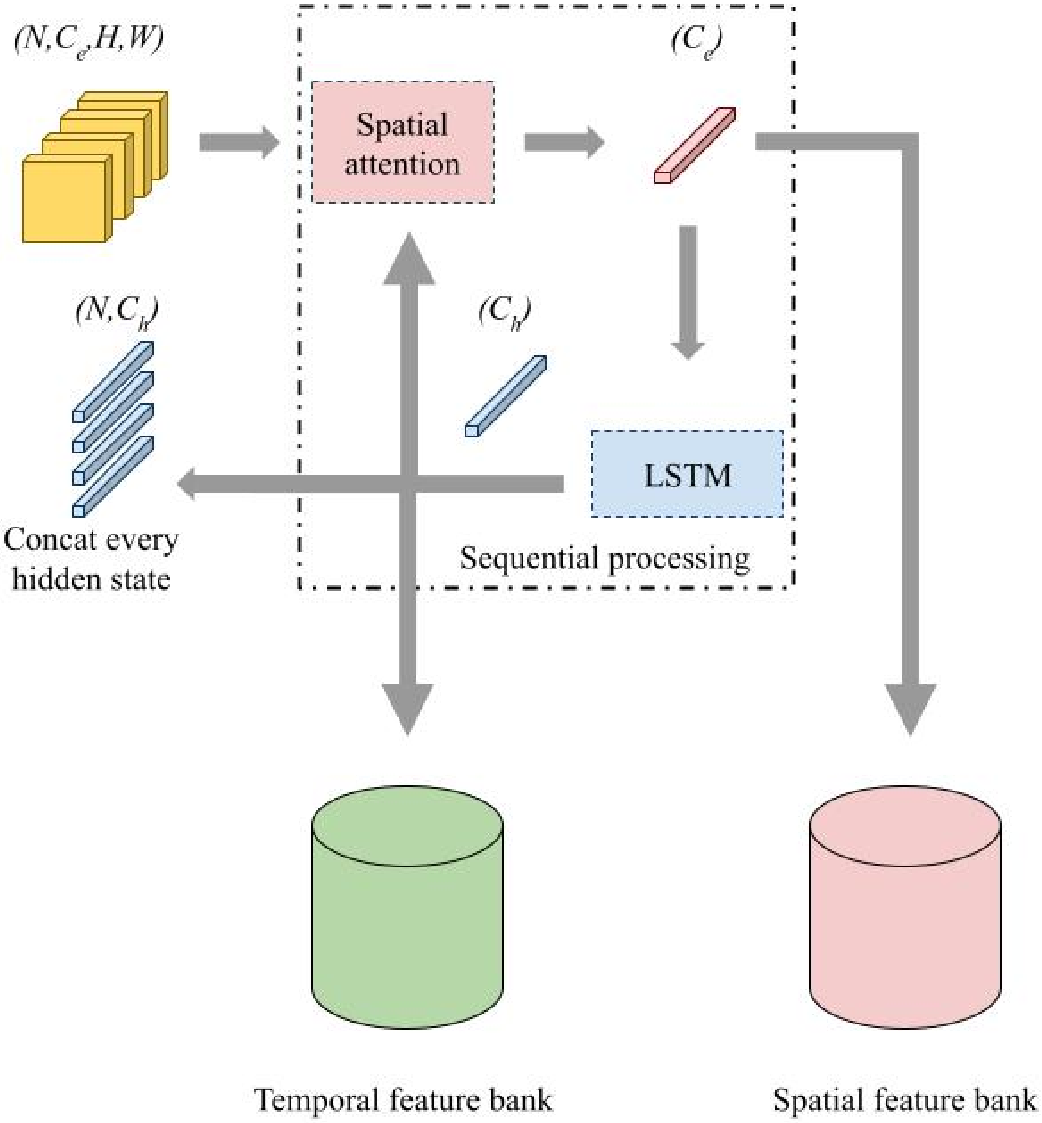

Figure 3-4. An illustration of how the features are extracted. Both the attended spatial features and the temporal features can be extracted and stored on our disk.

Chapter 4

Experiments

We discuss all the details and results of our experiments. We first introduce the datasets used in our experiments. Next, we elaborate on how we process the data. Then, we provide the details of how we implemented the model, the training, and the experiments. In the next two sections, we discuss the ablation study and external validation designs and results. Finally, we show how we visualize the learned spatial attention map and the temporal attention map.

4.1 Datasets

In this section, we first introduce the source and the target video datasets we use in our experiments. Next, we discuss the differences between the source dataset and the target dataset. The target dataset is used to evaluate the model's performance in external validation and domain adaptation. Finally, we discuss the spatial features dataset and the temporal features dataset.

4.1.1 Source Dataset

We used the dataset described in [66]. The dataset contains 99 videos of capsulorhexis, which is a critical step in cataract surgery. The videos were captured from the operating microscope, and processed to have a resolution of 640*480 and a frame rate of 59 frames per second. An expert surgeon assigned ground truth ratings for skill using the International Council of Ophthalmology's Surgical Competency Assessment Rubric-Phacoemulsification (ICO-OSCAR:phaco) [7]. Using the scores on the two items for capsulorhexis in ICO-OSCAR:phaco, which are rated on a Likert scale ranging from 2 to 5, we assigned videos an expert label if the score on at least one of the items was a 5 and the score on the other item was at least a 4, and a novice label if these criteria were not met. The dataset included 51 expert and 48 novice videos, which were evenly distributed among five folds for cross-validation. Every 12 frames in the videos, we manually annotated instrument tips and the points of entry of instruments into the eye. This paragraph is reproduced from our previous work [64].

4.1.2 Target Dataset

The target dataset is a newly collected dataset consisting of 51 videos of capsulorhexis. The recorded procedure, frame rate, and resolution are identical to the 99 videos dataset. The skill level labeling of the videos is also the same as the 99 videos dataset. However, there are no annotations of instrument tip locations. There are 5 expert-level videos and 46 novice-level videos. We use this dataset as the external dataset or the target domain for external validation and domain adaptation.

4.1.3 Comparison between the source dataset and the target dataset

The source and the target datasets are very different. There are covariate shifts and label shifts [72]. Figure 4-1 shows some sampled images from the videos from the two datasets. The figure shows the appearance differences between the images from the two datasets are not significant.

Table 4-Ishows there is a significant covariate shift between the two datasets. The mean, minimum, and maximum length of the videos from the target dataset are almost 3 times longer than those in the source dataset. Furthermore, the standard deviation of the videos in the target dataset is also almost 3 times larger than those in the source dataset. Moreover, videos from the target dataset show procedures performed by residents, who are trainees. Their moves are slower than experts. Videos from the source dataset are performed by expert surgeons and residents. Movements by expert surgeons tend to be much faster than that of the residents. The differences in the speed of movements and lengths of the videos between the two datasets are the main contributors to the covariate shift.

Table 4-II shows there is a significant label shift between the two datasets. The target dataset is scarce in positive samples. This could make domain adaptation especially difficult, as there might not be sufficient positive samples from the target domain to include in the source domain for semi-supervised domain adaptation.

	Mean	Std	Min	Max
Source	8677	6208	1712	29979
Target	20790	17080	5162	83389

Table 4-I. Statistics of the lengths of the videos from the source dataset and the target dataset. The numbers are the numbers of frames.

	Number of videos	Expert	Novice	Performer
Source	99	50	49	Expert surgeons and residents.
Target	51	5	46	Residents only.

Table 4-II. Table of class labels distribution statistics for the source and the target datasets.

4.1.4 Spatial feature dataset and temporal feature dataset

We extract features of videos from the source dataset using pretrained supervised attention model with the aggregation scheme, described in section 3.1 and section

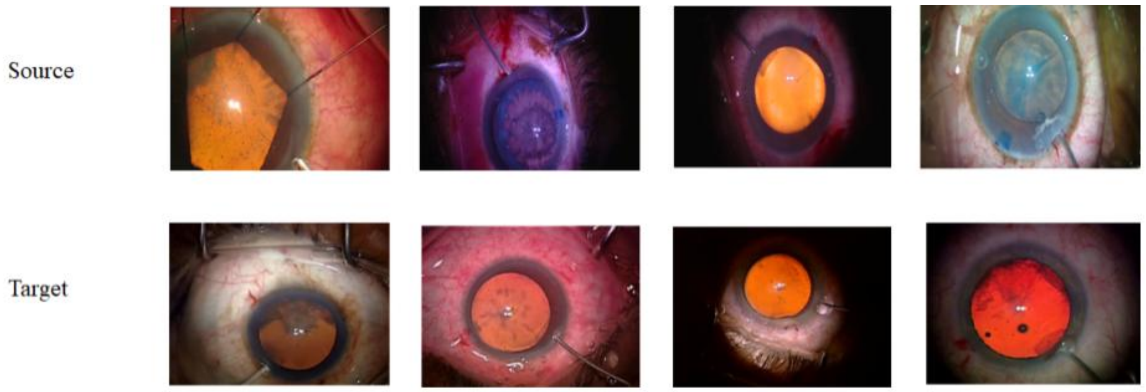

Figure 4-1. Images sampled from the source dataset and the target dataset.

3.2.3. We chose to use the aggregation scheme because it shows better performance in the external validation experiments, as shown in table 4-VI. Since the model is trained using five-fold cross-validation, for each test fold split, we extract features using the best model for each test fold.

The spatial feature dataset consists of spatial features of each frame of the videos. The spatial feature is the attended image feature, $\mathbf{x}_i^{att}$, discussed in section 3.2.3. The temporal feature dataset is composed of temporal features in each frame of the videos. The temporal feature is the hidden state from the LSTM of each frame.

4.2 Data processing

In this section, we introduce how the data are processed. We first discuss the dataset preparation process. Next, we discuss the sampling strategy for sampling images from the videos. Finally, we show how we augment the data to enrich the dataset.

4.2.1 Dataset preparation

The videos from both the source dataset and the target dataset are first anonymized. Then we use FFMPEG [73] to extract images from the videos. The source dataset is split into 5 folds for cross-fold validation.

The extracted features are stored in Numpy [74] arrays. Each Numpy array corresponds to the extracted features of one video.

4.2.2 Sampling

Sampling is extremely important for video-based deep learning approaches because it is impractical to train a complex neural network using all frames from the videos. It is simply computationally impractical. Therefore, it is necessary to drop most of the frames in the videos. However, with proper sampling, the model is able to learn from the sampled subset of video frames because of the high redundancy of information between consecutive frames.

We follow the sampling strategy from [25] for the source dataset and the target dataset. For each video, we sample a total of 256 frames from the video. We uniformly and randomly select a starting frame. Next, we select one frame every 8 frames until we have selected 256 frames. With the sampling interval set as 8, we ensure the loss of information is limited within the time window of the 256 frames. Since we are randomly selecting a starting frame, each time we sample frames from this video, the sampled clip should be different with a high probability. Furthermore, since for each training epoch, we sample a different clip from the video, after a sufficient amount of training epochs, all the sampled clips in all the epochs should sparsely cover most of the whole video. During testing, for each video, we sample three times and test the model with the three sampled clips. We use the averaged results as the final prediction.

For the feature datasets, we also sample 1 frame every 8 frames. However, we don't set a fixed amount of frames to be sampled. We sample from the start of the video to the end. Although this will introduce shifts between videos' lengths, [69] shows that using the whole video is beneficial.

4.2.3 Data augmentation

Data augmentation is an important technique to increase the amount of training data. In our case, the size of both the source and the target datasets are small. We use random crop, color jittering, horizontal flip, and random rotation to augment images from the videos. For the source dataset, the instrument tip locations are augmented with the corresponding images. We use the Albumentation 1.01 framework [75] for the augmentations.

We didn't apply any data augmentation for the feature datasets, as it is not straightforward to augment features.

4.3 Implementation details

Frameworks We use the PyTorch 1.3 framework [76] for implementation, training, and evaluation.

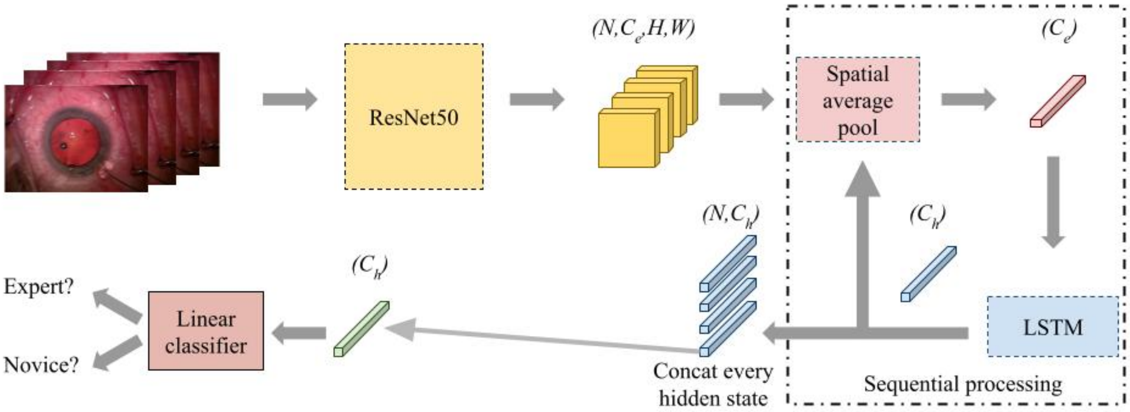

Figure 4-2. The network architecture of the no attention model.

Models We implemented 8 models in our experiments.

1. Baseline: we use the spatial-temporal attention model proposed by [25] as our baseline model. It uses the same structure shown in Fig3-1. Its spatial attention is not supervised. It uses the aggregation scheme. It does not use multi-task

learning.

2. Aggregation: supervised spatial attention model using the aggregation scheme (section 3.2.3).

3. Selection: supervised spatial attention model using the selection scheme (section 3.2.4).

4. Multi-task learning: Learning keypoint localization together with skill labels. The attention map is not explicitly supervised.

5. No attention model: spatial attention and temporal attention are removed from the baseline model. Figure 4-2 illustrates the network architecture.

6. No spatial attention model (No spatial): only removed spatial attention module from the aggregation model.

7. No temporal attention model (No temporal): only removed temporal attention module from the aggregation model.

8. Temporal network: temporal network built on top of the extracted features (section 3.4.2).

In the following text, we refer to models 1-4 as the attention models, and models 5-7 as the no attention models.

Training attention models We used the Adam optimizer with an initial learning rate of 1e-3 and decreased it by a factor of 10 as validation loss plateaued. The batch size is set as 2 for all models except for the multi-task learning model, for which it was set to 1 due to computational constraints. The ResNet-50 backbone was pre-trained on ImageNet[77] and frozen for training. We set the C_e as 2048 and C_h as 1024. The dimensions of $\mathbf{M}_e$, $\mathbf{M}_p$, and $\mathbf{M}_h$, are (1024×1), (1024×1024), and (2048×1024), respectively. The final linear classification layer (dimensions = $(1024, 1)$) is followed by the sigmoid function. All hyper-parameters are tuned empirically.

Training temporal network We use the same optimizer with the same hyper-parameters as those used for training attention models. The batch size is set as 20. The N_f is the number of sampled frames from the video, which is variable depending on the length of the video. C_f and C_{hf} are all set as 1024. The final linear classification layer is the same as that used for training attention models.

Training for semi-supervised domain adaptation The optimizer and all other hyper-parameters are the same as those used for training attention models. The only difference in training is that we also optimize a learnable adversarial parameter $w(z = source)$ that maximizes the loss term when training the Group-DRSL and Weighted-Group-DRSL models. Details are provided in the section 3.5.3 and section 3.5.4. This parameter is trained every 5 epochs. We train the models on the source domain and test them on the target domain. When training, we include 10 samples from the target domain, with 9 negative samples and 1 positive sample.

Statistical evaluation We computed accuracy, sensitivity, specificity, the receiver operating characteristic curve (ROC), and the area under it (AUC). To compute 95% confidence intervals, we used the Wilson method for sensitivity and specificity, and the DeLong method for AUC [78].

4.4 Ablation study

4.4.1 Comparison of the supervised attention with the unsupervised attention models

Table 4-III shows estimates and 95% confidence intervals of accuracy, sensitivity, specificity, and AUC for supervised attention models and the baseline models evaluated on the source dataset. AUC estimates show that the selection and aggregation models

Table 4-III. Estimates of performance and 95% confidence intervals for supervised attention models and the baseline models evaluated on the source dataset.

Model	Accuracy	Sensitivity	Specificity	AUC
Baseline	0.73 (0.57 to 0.89)	0.76 (0.63 to 0.86)	0.69 (0.55 to 0.80)	0.78 (0.69 to 0.87)
Baseline†	-	0.84 (0.72 to 0.92)	0.75 (0.61 to 0.85)	0.78 (0.69 to 0.88)
Multi-task	0.70 (0.48 to 0.92)	0.67 (0.53 to 0.78)	0.73 (0.59 to 0.83)	0.73 (0.63 to 0.83)
Selection	0.77 (0.59 to 0.95)	0.73 (0.59 to 0.83)	0.81 (0.68 to 0.90)	0.85 (0.78 to 0.93)
Aggregation	0.79 (0.51 to 1.00)	0.76 (0.63 to 0.86)	0.79 (0.66 to 0.88)	0.85 (0.77 to 0.93)

†: Prior results reported in [25]. The model in [25] has the same structure as our baseline model but uses ResNet-101 as the backbone. The model was trained using 64 frames per batch with a batch size of one. The dimensions of $\mathbf{M}_e$, $\mathbf{M}_p$, and $\mathbf{M}_h$ were set to (256×1), (256×256), and (2048×256), respectively. Every fourth frame was sampled.

with supervised attention have more discrimination than a multi-task learning model and a baseline unsupervised attention model. Compared with the baseline model the selection and aggregation models have similar estimates of sensitivity and higher estimates of specificity. Estimates for the multi-task learning model are lower than those for the other models, which we postulate is because of insufficient amounts of training data. The results prove our hypothesis that supervising spatial attention helps improve model performance.

Supervised attention with the selection scheme performs similarly to the aggregation scheme in table 4-III. The selection scheme selects the features of one pixel in the feature map generated by the CNN backbone (ResNet-50), whereas the aggregation scheme aggregates the feature map. The observation that they perform similarly shows it is sufficient to achieve high performance using only a small amount of information in each frame.

Table 4-IV. Estimates of performance and 95% confidence intervals for attention models and the no attention models evaluated on the source dataset. The results of the Baseline models, and the Aggregation model from table 4-III are repeated in here for clearer comparison. The results of the Selection model is omitted as the no attention models are variants from the Aggregation model and the Baseline models only. We chose the Aggregation model rather than the Selection model because it shows better performance in the external validation experiments.

Model	Accuracy	Sensitivity	Specificity	AUC
Baseline	0.73 (0.57 to 0.89)	0.76 (0.63 to 0.86)	0.69 (0.55 to 0.80)	0.78 (0.69 to 0.87)
Baseline†	-	0.84 (0.72 to 0.92)	0.75 (0.61 to 0.85)	0.78 (0.69 to 0.88)
No temporal	0.81 (0.72 to 0.87)	0.78 (0.65 to 0.88)	0.83 (0.70 to 0.91)	0.88 (0.81 to 0.95)
No spatial	0.79 (0.70 to 0.86)	0.78 (0.65 to 0.88)	0.79 (0.66 to 0.88)	0.88 (0.81 to 0.95)
No attention	0.81 (0.72 to 0.87)	0.78 (0.65 to 0.88)	0.83 (0.70 to 0.91))	0.87 (0.80 to 0.94)
Aggregation	0.79 (0.51 to 1.00)	0.76 (0.63 to 0.86)	0.79 (0.66 to 0.88)	0.85 (0.77 to 0.93)

†: Prior results reported in [25]. The model in [25] has the same structure as our baseline model but uses ResNet-101 as the backbone. The model was trained using 64 frames per batch with a batch size of one. The dimensions of $\mathbf{M}_e$, $\mathbf{M}_p$, and $\mathbf{M}_h$ were set to (256×1), (256×256), and (2048×256), respectively. Every fourth frame was sampled.

Table 4-V. Estimates of performance and 95% confidence intervals for temporal networks trained on the spatial feature dataset and the temporal feature dataset.

Dataset	Accuracy	Sensitivity	Specificity	AUC
Spatial features	0.67 (0.57 to 0.75)	0.63 (0.49 to 0.75)	0.71 (0.57 to 0.82)	0.73 (0.63 to 0.83)
Temporal features	0.73 (0.63 to 0.81)	0.75 (0.61 to 0.84)	0.71 (0.57 to 0.82)	0.77 (0.68 to 0.86)

4.4.2 Comparison between attention models with no attention models

Table 4-IV shows results for no attention models. Contrary to what [25] found, our experiments show not using temporal attention and only using supervised spatial attention (the no temporal model) performs the best among all other models. Furthermore, not using spatial attention and only using temporal attention (the no spatial model in the table) and not using any attention (no attention model) perform similarly to the no temporal model. Not using any attention also performs much better than the baseline model, which uses unsupervised spatial attention and temporal attention. These results show using without supervising the attention with additional signals such as instrument locations, the model tends to over-fit the training dataset. Our finding is contrary to other previous works, which claim using attention help improve the model's performance and generalizability [25, 33, 39, 49]. We hypothesize that these surprising results are caused by insufficient data in the source dataset. Adding additional parameters through attention modules increases the complexity of the models, which requires additional signals or data to avoid over-fitting. This would explain why supervising spatial attention improves performance.

4.4.3 Comparison between spatial features and temporal features for training the temporal attention network

Table 4-V shows results for temporal networks trained on the spatial feature dataset and the temporal feature dataset. Their performances are worse than those of the

Table 4-VI. Estimates of performance and 95% confidence intervals for supervised, unsupervised attention models, and the no attention model externally evaluated on the target dataset.

Model	Accuracy	Sensitivity	Specificity	AUC
Baseline	0.14 (0.07 to 0.26)	0.60 (0.23 to 0.88)	0.09 (0.03 to 0.20)	0.37 (0.06 to 0.68)
No attention	0.37 (0.25 to 0.51)	0.60 (0.23 to 0.88)	0.35 (0.23 to 0.49)	0.47 (0.11 to 0.83)
Selection	0.41 (0.29 to 0.55)	0.40 (0.12 to 0.77)	0.41 (0.28 to 0.56)	0.45 (0.23 to 0.67)
Aggregation	0.16 (0.08 to 0.28)	0.07 (0.02 to 0.18)	1.00 (0.57 to 1.00)	0.50 (0.13 to 0.88)

attention models as they were not trained end-to-end. Using temporal features to train the temporal network gives better performance than using spatial features. Therefore, we chose to use temporal features to generate temporal attention.

4.5 External validation and domain adaptation

Table 4-VI shows the results of our external validation results. These results show supervising the spatial attention and using temporal attention with the aggregation scheme performs best in external validation. Contrary to the results in internal validation shown in table 4-IV, the supervised spatial attention model with the aggregation scheme outperforms the no attention model. It further proves that supervising spatial attention improves the model's generalizability.

Table 4-VII shows results for semi-supervised domain adaptation. We implemented the adaptation methods with the baseline model and the no attention model. Since instrument tips in the target domain, the target dataset, are not annotated, we couldn't apply these adaptation methods with the supervised spatial attention models. The results show for all adaptation methods, the baseline model that uses attention performs worse than the no attention model. This observation again shows using attention causes the model to over-fit the training data. The Group-DRSL method

Table 4-VII. Estimates of performance and 95% confidence intervals for semi-supervised domain adaptation methods with the unsupervised attention model and the no attention model. They were externally evaluated on the target dataset.

Method	Accuracy	Sensitivity	Specificity	AUC
Baseline model				
None	0.65 (0.47 to 0.79)	0.00 (0.00 to 0.56)	0.71 (0.53 to 0.85)	0.33 (-0.03 to 0.70)
Vanilla SSDA	0.71 (0.53 to 0.84)	0.00 (0.00 to 0.56)	0.79 (0.60 to 0.90)	0.24 (0.05 to 0.42)
Group-DRSL	0.77 (0.60 to 0.89)	0.00 (0.00 to 0.56)	0.86 (0.69 to 0.94)	0.45 (0.12 to 0.78)
Weighted-Group-DRSL	0.87 (0.71 to 0.95)	0.00 (0.00 to 0.56)	0.96 (0.82 to 0.99)	0.32 (-0.02 to 0.66)
No attention model				
None	0.61 (0.44 to 0.76)	0.33 (0.06 to 0.79)	0.64 (0.46 to 0.79)	0.42 (-0.06 to 0.90)
Vanilla SSDA	0.77 (0.60 to 0.89)	0.33 (0.06 to 0.79)	0.82 (0.64 to 0.92)	0.44 (-0.02 to 0.90)
Group-DRSL	0.90 (0.75 to 0.97)	0.33 (0.06 to 0.79)	0.96 (0.82 to 0.99)	0.55 (0.12 to 0.98)
Weighted-Group-DRSL	0.87 (0.71 to 0.95)	0.00 (0.00 to 0.56)	0.96 (0.82 to 0.99)	0.52 (0.06 to 0.99)

consistently and significantly improves both models' performances. The Weighted-Group-DRSL method shows less than expected results. We hypothesize that the reason behind this result is the lack of positive samples included from the target domain. We could only include 1 positive sample from the target domain because there are only 5 positive samples in the target domain in total. Despite assigning higher weights, the model couldn't learn a sufficient amount of transferable knowledge from the 1 positive sample. The scarcity of positive samples may also be the cause of why all the models' sensitivity scores are low.

4.6 Visualization

We visualized the spatial attention map and temporal attention map generated by the supervised spatial attention model with the aggregation scheme discussed in section 3.2.3. Figure 1-1 shows the results. The attention map is colored with the Jet colormap, where red and bright colors denote high attention value. The first two rows show spatial attention maps extracted from three test videos. These visualizations are upsampled spatial attention maps overlaid on the input images. The original spatial attention maps are highly localized, and usually, only one pixel in the spatial attention map is activated. With these maps, only the critical information around the instrument tips is used for predictions. The highly localized attention map together with the improvements shown in table 4-III and table 4-VI empirically supports the hypothesis that supervising attention with instrument tip trajectories improves performance by regularizing the network to use the most relevant information to assess surgical skill.

Figure 4-3 shows the extracted temporal attention from the temporal network. This temporal attention map is normalized across the whole video. We examined temporal attention maps generated from several videos with experts. They claim the learned temporal attention maps are not easily explainable compared with the generated spatial attention maps. Future work may focus on how to generate more

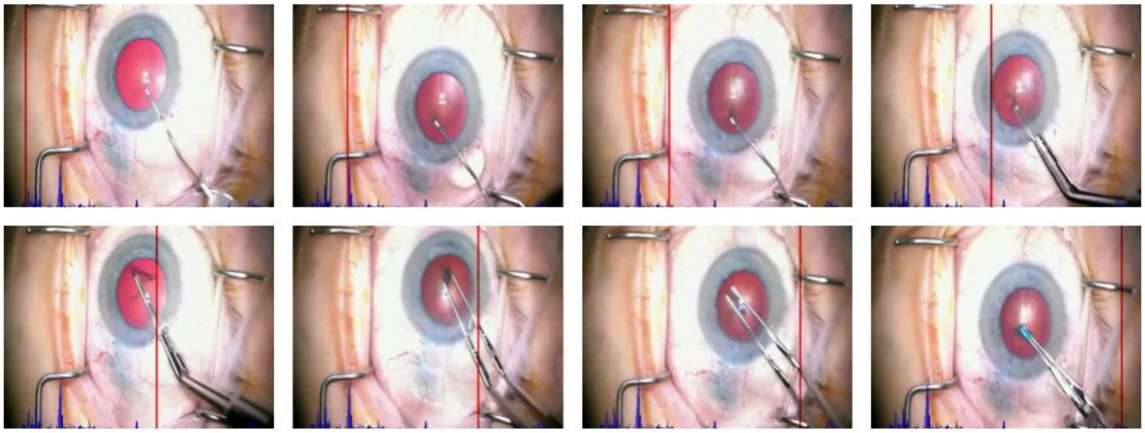

Figure 4-3. Extracted temporal attention of the whole video. The heights of the blue bars at the bottom of the images show how important the corresponding frame is. The red bar shows the progress of the video and locates the current frame.

meaningful and explainable temporal attention maps.

Conclusions and discussion

This thesis first describes a method to explicitly supervise the spatial attention model using ancillary information inherent within the images that are known to correlate with surgical skill (prediction target). Supervising the spatial attention map with relevant ancillary information may result in more accurate models than a multi-task learning approach. Our experiment results support this claim. Furthermore, our results show supervising the spatial attention map help mitigate the problem of overfitting and improves generalizability in external validation. This thesis also presents a two-stage method for generating temporal attention for whole videos. Finally, we explored several semi-supervised domain adaptation techniques to improve the model performance on external validation. We found our Group-DRSL-[70] based method performs the best.

References

1. Weiser, T. G. *et al.* Estimate of the global volume of surgery in 2012: an assessment supporting improved health outcomes. *The Lancet* **385,** S11 (2015).
2. Birkmeyer, J. D. *et al.* Surgical skill and complication rates after bariatric surgery. *New England Journal of Medicine* **369,** 1434–1442 (2013).
3. Reznick, R. K. Teaching and testing technical skills. *The American journal of surgery* **165,** 358–361 (1993).
4. Cremers, S. L., Ciolino, J. B., Ferrufino-Ponce, Z. K. & Henderson, B. A. Objective assessment of skills in intraocular surgery (OASIS). *Ophthalmology* **112,** 1236–1241 (2005).
5. Cremers, S. L., Lora, A. N. & Ferrufino-Ponce, Z. K. Global rating assessment of skills in intraocular surgery (GRASIS). *Ophthalmology* **112,** 1655–1660 (2005).
6. Gauba, V. *et al.* Human reliability analysis of cataract surgery. *Archives of ophthalmology* **126,** 173–177 (2008).
7. Golnik, K. C. *et al.* Cataract surgical skill assessment. *Ophthalmology* **118,** 427–427 (2011).
8. Saleh, G. M. *et al.* Objective structured assessment of cataract surgical skill. *Archives of ophthalmology* **125,** 363–366 (2007).
9. Martin, J. *et al.* Objective structured assessment of technical skill (OSATS) for surgical residents. *British journal of surgery* **84,** 273–278 (1997).
10. Wang, C. *et al.* Crowdsourcing in health and medical research: a systematic review. *Infectious diseases of poverty* **9,** 1–9 (2020).
11. Malpani, A., Vedula, S. S., Chen, C. C. G. & Hager, G. D. A study of crowdsourced segment-level surgical skill assessment using pairwise rankings. *International journal of computer assisted radiology and surgery* **10,** 1435–1447 (2015).
12. Baghdadi, A., Hussein, A. A., Ahmed, Y., Cavuoto, L. A. & Guru, K. A. A computer vision technique for automated assessment of surgical performance using surgeons' console-feed videos. *International journal of computer assisted radiology and surgery* **14,** 697–707 (2019).
13. Vedula, S. S., Ishii, M. & Hager, G. D. Objective assessment of surgical technical skill and competency in the operating room. *Annual review of biomedical engineering* **19,** 301–325 (2017).
14. Maier-Hein, L. *et al.* Surgical data science for next-generation interventions. *Nature Biomedical Engineering* **1,** 691–696 (2017).

15. Maier-Hein, L. *et al.* Surgical data science–from concepts toward clinical translation. *Medical image analysis* **76,** 102306 (2022).

16. Vedula, S. S. & Hager, G. D. Surgical data science: the new knowledge domain. *Innovative surgical sciences* **2,** 109–121 (2017).

17. Law, H., Ghani, K. & Deng, J. *Surgeon Technical Skill Assessment using Computer Vision based Analysis* in *Proceedings of the 2nd Machine Learning for Healthcare Conference* (eds Doshi-Velez, F. *et al.*) **68** (PMLR, 18–19 Aug 2017), 88–99.

18. Lin, S., Qin, F., Bly, R. A., Moe, K. S. & Hannaford, B. *Automatic Sinus Surgery Skill Assessment Based on Instrument Segmentation and Tracking in Endoscopic Video* in *Multiscale Multimodal Medical Imaging* (eds Li, Q., Leahy, R., Dong, B. & Li, X.) (Springer International Publishing, Cham, 2020), 93–100.

19. Liu, D. *et al. Surgical Skill Assessment on In-Vivo Clinical Data via the Clearness of Operating Field* in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019* (eds Shen, D. *et al.*) (Springer International Publishing, Cham, 2019), 476–484.

20. Lavanchy, J. L. *et al.* Automation of surgical skill assessment using a three-stage machine learning algorithm. *Scientific reports* **11,** 1–9 (2021).

21. Jin, A. *et al. 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), 691–699.

22. Liu, D. *et al. Towards Unified Surgical Skill Assessment* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (June 2021), 9522–9531.

23. Kitaguchi, D. *et al.* Development and Validation of a 3-Dimensional Convolutional Neural Network for Automatic Surgical Skill Assessment Based on Spatiotemporal Video Analysis. *JAMA Network Open* **4,** e2120786–e2120786. eprint: https://jamanetwork.com/journals/jamanetworkopen/articlepdf/2782991/kitaguchi_2021_oi_210615_1628614594.81153.pdf (Aug. 2021).

24. Carreira, J. & Zisserman, A. *Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset* in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (July 2017).

25. Hira, S. *et al. Video-based assessment of intraoperative surgical skill* unpublished. 2022.

26. Guo, M.-H. *et al.* Attention mechanisms in computer vision: A survey. *Computational Visual Media,* 1–38 (2022).

27. Hu, J., Shen, L. & Sun, G. *Squeeze-and-excitation networks* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), 7132–7141.

28. Woo, S., Park, J., Lee, J.-Y. & Kweon, I. S. *Cbam: Convolutional block attention module* in *Proceedings of the European conference on computer vision (ECCV)* (2018), 3–19.

29. Dai, J. *et al. Deformable convolutional networks* in *Proceedings of the IEEE international conference on computer vision* (2017), 764–773.

30. Zheng, M. *et al.* End-to-end object detection with adaptive clustering transformer. *arXiv preprint arXiv:2011.09315* (2020).

31. Wang, X., Girshick, R., Gupta, A. & He, K. *Non-local neural networks* in *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), 7794–7803.

32. Du, W., Wang, Y. & Qiao, Y. Recurrent spatial-temporal attention network for action recognition in videos. *IEEE Transactions on Image Processing* **27,** 1347–1360 (2017).

33. Li, Z., Huang, Y., Cai, M. & Sato, Y. *Manipulation-skill assessment from videos with spatial attention network* in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops* (2019), 0–0.

34. Mason, J. D., Ansell, J., Warren, N. & Torkington, J. Is motion analysis a valid tool for assessing laparoscopic skill? *Surgical endoscopy* **27,** 1468–1477 (2013).

35. Ghasemloonia, A. *et al.* Surgical skill assessment using motion quality and smoothness. *Journal of surgical education* **74,** 295–305 (2017).

36. Yan, C. *et al.* STAT: Spatial-temporal attention mechanism for video captioning. *IEEE transactions on multimedia* **22,** 229–241 (2019).

37. Zia, A. & Essa, I. Automated surgical skill assessment in RMIS training. *International journal of computer assisted radiology and surgery* **13,** 731–739 (2018).

38. Bahdanau, D., Cho, K. H. & Bengio, Y. *Neural machine translation by jointly learning to align and translate* in *3rd International Conference on Learning Representations, ICLR 2015* (2015).

39. Chaudhari, S., Mithal, V., Polatkan, G. & Ramanath, R. An attentive survey of attention models. *ACM Transactions on Intelligent Systems and Technology (TIST)* **12,** 1–32 (2021).

40. Galassi, A., Lippi, M. & Torroni, P. Attention in natural language processing. *IEEE Transactions on Neural Networks and Learning Systems* **32,** 4291–4308 (2020).

41. Cho, K., Courville, A. & Bengio, Y. Describing multimedia content using attention-based encoder-decoder networks. *IEEE Transactions on Multimedia* **17,** 1875–1886 (2015).

42. Wang, F. & Tax, D. M. Survey on the attention based RNN model and its applications in computer vision. *arXiv preprint arXiv:1601.06823* (2016).

43. Xu, K. *et al. Show, attend and tell: Neural image caption generation with visual attention* in *International conference on machine learning* (2015), 2048–2057.

44. Luong, T., Pham, H. & Manning, C. D. *Effective Approaches to Attention-based Neural Machine Translation* in *EMNLP* (2015).

45. Vaswani, A. *et al.* Attention is all you need. *Advances in neural information processing systems* **30** (2017).

46. Dosovitskiy, A. *et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale* in *International Conference on Learning Representations* (2020).

47. Das, A., Agrawal, H., Zitnick, L., Parikh, D. & Batra, D. Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions? *Computer Vision and Image Understanding* **163.** Language in Vision, 90–100 (2017).

48. Gan, C., Li, Y., Li, H., Sun, C. & Gong, B. *VQS: Linking Segmentations to Questions and Answers for Supervised Attention in VQA and Question-Focused Semantic Segmentation* in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (Oct. 2017).

49. Liu, L., Utiyama, M., Finch, A. & Sumita, E. *Neural Machine Translation with Supervised Attention* in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers* (The COLING 2016 Organizing Committee, Osaka, Japan, Dec. 2016), 3093–3102.

50. Motiian, S., Jones, Q., Iranmanesh, S. & Doretto, G. Few-shot adversarial domain adaptation. *Advances in neural information processing systems* **30** (2017).

51. Choi, J., Sharma, G., Chandraker, M. & Huang, J.-B. *Unsupervised and semi-supervised domain adaptation for action recognition from drones* in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (2020), 1717–1726.

52. Li, B. *et al. Learning invariant representations and risks for semi-supervised domain adaptation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 1104–1113.

53. Saito, K., Kim, D., Sclaroff, S., Darrell, T. & Saenko, K. *Semi-supervised domain adaptation via minimax entropy* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), 8050–8058.

54. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B. & Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research* **13,** 723–773 (2012).

55. Long, M., Cao, Y., Wang, J. & Jordan, M. *Learning transferable features with deep adaptation networks* in *International conference on machine learning* (2015), 97–105.

56. Ganin, Y. & Lempitsky, V. *Unsupervised domain adaptation by backpropagation* in *International conference on machine learning* (2015), 1180–1189.

57. Kim, D. *et al.* Cross-domain self-supervised learning for domain adaptation with few source labels. *arXiv preprint arXiv:2003.08264* (2020).

58. Yue, X. *et al. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 13834–13844.

59. Gong, B., Shi, Y., Sha, F. & Grauman, K. *Geodesic flow kernel for unsupervised domain adaptation* in *2012 IEEE conference on computer vision and pattern recognition* (2012), 2066–2073.

60. Luo, Z., Zou, Y., Hoffman, J. & Fei-Fei, L. F. Label efficient learning of transferable representations acrosss domains and tasks. *Advances in neural information processing systems* **30** (2017).

61. Sohn, K. *et al. Unsupervised domain adaptation for face recognition in unlabeled videos* in *Proceedings of the IEEE International Conference on Computer Vision* (2017), 3210–3218.

62. Chen, M.-H. *et al. Temporal attentive alignment for large-scale video domain adaptation* in *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2019), 6321–6330.

63. Jamal, A., Namboodiri, V. P., Deodhare, D. & Venkatesh, K. *Deep Domain Adaptation in Action Space.* in *BMVC* **2** (2018), 5.

64. Wan, B., Peven, M., Sikder, S., Hager, G. & Vedula, S. S. *Supervised attention for video-based assessment of intraoperative surgical skill* in *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* submitted February 2022. (Springer International Publishing, 18–22 Sept. 2022).

65. Jian, Z. *et al. Multitask learning for video-based surgical skill assessment* in *2020 Digital Image Computing: Techniques and Applications (DICTA)* (2020), 1–8.

66. Kim, T. S. *et al.* Objective assessment of intraoperative technical skill in capsulorhexis using videos of cataract surgery. *International journal of computer assisted radiology and surgery* **14,** 1097–1105 (2019).

67. Newell, A., Huang, Z. & Deng, J. Associative embedding: End-to-end learning for joint detection and grouping. *Advances in neural information processing systems* **30** (2017).

68. Wang, L. *et al. Temporal segment networks: Towards good practices for deep action recognition* in *European conference on computer vision* (2016), 20–36.

69. Liu, X., Pintea, S. L., Nejadasl, F. K., Booij, O. & van Gemert, J. C. *No frame left behind: Full video action recognition* in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), 14892–14901.

70. Hu, W., Niu, G., Sato, I. & Sugiyama, M. *Does distributionally robust supervised learning give robust classifiers?* in *International Conference on Machine Learning* (2018), 2029–2037.

71. Ganin, Y. *et al.* Domain-adversarial training of neural networks. *The journal of machine learning research* **17,** 2096–2030 (2016).

72. Storkey, A. in *Dataset Shift in Machine Learning* (eds Quionero-Candela, J., Sugiyama, M., Schwaighofer, A. & Lawrence, N. D.) 3–28 (The MIT Press, Cambridge, Mass, 2008).

73. Popinet, S. *GTS: GNU Triangulated Surface library* http://gts.sourceforge.net/. 2000–2004.

74. Harris, C. R. *et al.* Array programming with NumPy. *Nature* **585,** 357–362 (Sept. 2020).

75. Buslaev, A. *et al.* Albumentations: fast and flexible image augmentations. *Information* **11,** 125 (2020).

76. Paszke, A. *et al.* in *Advances in Neural Information Processing Systems 32* 8024–8035 (Curran Associates, Inc., 2019).

77. Deng, J. *et al. Imagenet: A large-scale hierarchical image database* in *2009 IEEE conference on computer vision and pattern recognition* (2009), 248–255.

78. DeLong, E. R., DeLong, D. M. & Clarke-Pearson, D. L. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics,* 837–845 (1988).

BOHUA WAN

https://github.com/GlenGGG ◇ https://bohua-wan.netlify.app

EDUCATION

Johns Hopkins University *January 2021 - Present*
Whiting School of Engineering

- M.S.E. in Computer Science, expected June 2022 **GPA: 4.0**

China University of Petroleum, Beijing *September 2016 - June 2020*
College of Information Science and Engineering

- B.S. in Computer Science and Technology **Rank: 1/103** **GPA: 91.9/100.0**

RESEARCH EXPERIENCE

Spatial-temporal attention for video-based assessment of intraopertive surgical skills. *Oct 2021 - Present*
Research Assistant–supervised by Professor Gregory D. Hager *Johns Hopkins University*

- Proposed a model that supervises the spatial attention map with instrument tip locations. Experiments show improved performance and generalizability in internal and external validations.
- Proposed a two-stage method that generates temporal attention for the whole video instead of short clips.
- Evaluated several semi-supervised domain adaptation techniques for improving models' performance in external validations.

Framework to unify statistical and machine learning concepts on the generalizability of CPMs *July 2021 - Present*
Research Assistant–supervised by Professor Swaroop Vedula *Johns Hopkins University*

- Matched statistical concepts to dataset shifts in machine learning literature. For example, measurement bias maps to domain shift, and sampling bias maps to sample selection shift.
- Use causal inference and selection diagrams to analyze the transportability of relations under dataset shifts.
- Proposed a hypothesis that describes a condition under which no dataset shit may exist. By breaking this hypothesis, we have the root causes of all dataset shifts.
- Designed a checklist for medical researchers to assess the risk of dataset shifts between the development dataset and the application data.

Unsupervised Domain Adaptation for Image Classification *April 2021 - May 2021*
Research team lead *course project at Johns Hopkins University*

- Investigated Adversarial Discriminative Domain Adaptation (ADDA) and Deep CORAL methods with 3 other group members.
- Proposed a new network by using Deep CORAL to constrain ADDA from drastic deviating from the pretrained initialization, which is proven to be both effective and efficient in terms of performance and training time.

Skeleton-based human interaction recognition with graph convolutional network *September 2019 - January 2021*
Research Assistant–supervised by Professor Liping Zhu *China University of Petroleum, Beijing*

- Designed a Relational Adjacency Matrix to represent relational graphs between separate skeletons using geometric features and relative attention.
- Proposed Dyadic Relational Graph Convolutional Network, which achieves state-of-the-art accuracy on three challenging datasets with improvements of 6.63% on NTU-RGB+D and 5.47% on NTU-RGB+D 120 over the baseline model.
- Our methods consistently help advanced models achieve higher accuracy of 1.26% on NTU-RGB+D and 2.86% on NTU-RGB+D 120.

PUBLICATIONS

[1] *Zhu, L., **Wan, B. (Co-First)**, Li, C., Tian, G., Hou, Y., & Yuan, K. (2021). Dyadic relational graph convolutional networks for skeleton-based human interaction recognition. Pattern Recognition, 115, 107920.*

[2] ***Wan, B.**, Caffo B. & Vedula S. S. (2022) A Unified Framework on Generalizability of Clinical Prediction Models. Front. Artif. Intell. 5:872720. doi: 10.3389/frai.2022.872720*

SKILLS

Programming Languages	Python (proficient), C/C++ (proficient), Java (proficient), JavaScript (familiar)
Platforms & Frameworks	PyTorch (proficient), Django (proficient), Qt (proficient), Spark Java (proficient) CUDA (familiar), TensorFlow (familiar)
Deep Learning Techniques	CNN (proficient), GCN (proficient), GNN (familiar), GAN (familiar), DQN (familiar)

HONORS & AWARDS

CNPC Scholarship (exclusively rewarded to the top 5%, one of the highest honors in our university.)	*Fall, 2018*
Sinopec Scholarship (exclusively rewarded to the top 5%, one of the highest honors in our university.)	*Fall, 2017*

Biographical sketch

Bohua Wan is currently a second-year master's student studying Computer Science at Johns Hopkins University. He received his bachelor's degree in Computer Science and Technology from China University of Petroleum, Beijing, where he was supervised by Professor Liping Zhu. He is currently working with Professor Swaroop Vedula on dataset shift and medical video analysis. He is also fortunate to be advised by Professor Gregory Hager for his master's thesis, which is about spatial-temporal attention for video-based assessment of intraoperative surgical skills.

His research interests are in computer vision and machine learning. He is especially interested in explainable and generalizable machine learning. He has previous experience in action recognition, video understanding, spatial-temporal reasoning, domain adaptation, and reinforcement learning. His current research focus is on dataset shifts and spatial-temporal attention.