2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

daVinciNet: Joint Prediction of Motion and Surgical State in Robot-Assisted Surgery

Yidan Qin[1,2], Seyedshams Feyzabadi[1], Max Allan[1], Joel W. Burdick[2], Mahdi Azizian[1]

***Abstract*— This paper presents a technique to concurrently and jointly predict the future trajectories of surgical instruments and the future state(s) of surgical subtasks in robot-assisted surgeries (RAS) using multiple input sources. Such predictions are a necessary first step towards shared control and supervised autonomy of surgical subtasks. Minute-long surgical subtasks, such as suturing or ultrasound scanning, often have distinguishable tool kinematics and visual features, and can be described as a series of fine-grained states with transition schematics. We propose *daVinciNet* - an end-to-end dual-task model for robot motion and surgical state predictions. *daVinciNet* performs concurrent end-effector trajectory and surgical state predictions using features extracted from multiple data streams, including robot kinematics, endoscopic vision, and system events. We evaluate our proposed model on an extended Robotic Intra-Operative Ultrasound (RIOUS+) imaging dataset collected on a da Vinci® Xi surgical system and the JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS). Our model achieves up to 93.85% short-term (0.5s) and 82.11% long-term (2s) state prediction accuracy, as well as 1.07mm short-term and 5.62mm long-term trajectory prediction error.**

I. INTRODUCTION

The implementation of autonomy in the field of surgical robotics, from passive functionalities such as virtual fixtures [1] to autonomous surgical tasks [2], [3], has attracted the attention of many. Such systems enrich the manual teleoperation experience in robot-assisted surgeries (RAS) and assist the surgeons in many ways. Enhancements include automated changes in the user interface during surgery, additional surgeon-assisting system functionalities, and shared control or even autonomous tasks [4]–[6]. In 2016, Yang et al. proposed a definition of the levels of autonomy in medical robotics, ranging from mechanical robot guidance to fully autonomous surgical procedures [7], where the sensing of the user's desires play an integral role. One prerequisite for the applications mentioned above is the ability to anticipate the surgeon's intention and the robot's motions. Prediction of the robotic surgical instruments' trajectories, for instance, contributes to collision prediction and avoidance, including collisions between instruments or with obstacles in the proximity. It also has applications to safe multi-agent surgical systems where various surgical tasks are performed concurrently. Weede et al. presented an instrument trajectory prediction method for optimal endoscope positioning through autonomous endoscopic guidance [8]. The prediction of the next fine-grained surgical states, either

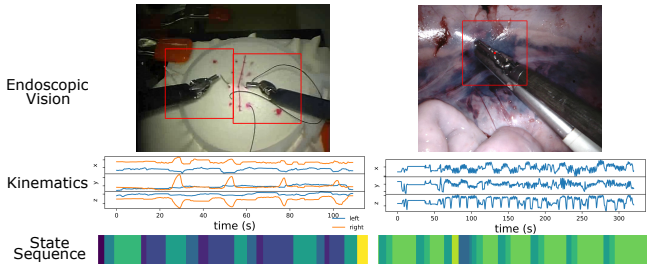

Fig. 1: Sample data from JIGSAWS (left) and RIOUS+ (right). The top row shows the endoscopic vision, with RoI bounding boxes (areas surrounding the end-effectors). The middle row shows the end-effector paths in Cartesian coordinates. The bottom row shows the state sequences.

fine-grained action states (picking up a needle) [9] or surgical phases (bladder dissection) [10], is useful in many surgeon-assisting features. Examples include the predictive triggering of cloud-based features or heavy-processing services which are inherently time consuming. This functionality provides a more seamless operational workflow. The prediction of the next surgical state also allows for more synchronized collaborations between the surgeons and operating room staff through workflow recognition [11], [12].

Prediction of a robots surgical instruments' motion during a surgical task has found applications in surgical instrument tracking [13], [14] and visual window tracking [15], [16]. Using only robot kinematics data, Staub et al. integrated surgical instrument trajectory and pose predictions with silhouette-based instrument tracking by using a Kalman filter to generate a pose prediction [13]. Similarly, instrument tracking procedures based on Kalman filtering with fusion of kinematics and visual data and visual matching of markers and appearance has been investigated [14]. Additionally, a hybrid grey prediction model was implemented for autonomous endoscope navigation during RAS [15]. Weede et al. proposed a guidance system for finding the optimal endoscope position through trajectory clustering and Markov Model (MM) for long-term instrument trajectory and optimal endoscope pose predictions [8]. These methods, however, all used surgical instrument motion prediction as auxiliary information to improve performance of their respective applications, without an extensive focus on improving motion prediction accuracy. Additionally, the prediction of surgical instrument trajectory seconds ahead has not been extensively researched. Weede et al. achieved long-term instrument trajectory prediction by trajectory clustering and aggregated

[1]Intuitive Surgical Inc., 1020 Kifer Road, Sunnyvale, CA,94086, USA
[2]Department of Mechanical and Civil Engineering, California Institute of Technology, 1200 E California Blvd, Pasadena, CA, 91125, USA
Emails: Ida.Qin@intusurg.com, Mahdi.Azizian@intusurg.com

time series data into categories [8]. Their MM-based model predicts the class of movements instead of the numerical temporal sequence of motion trajectory, which significantly limits its applications.

Surgical task such as suturing can be practically modeled as a Finite State Machine (FSM), with a list of discrete states (actions and non-actions) and possible transitions between states [17]. Classically, the task is formulated as a MM and the transition probability matrix is learned from data [8], [12], [18]–[20]. While powerful, Markov models do not capture temporal information, such as the duration of the last state and non-Markovian state sequences [8], [19]. With recent advances in machine learning, more learning-based state estimation methods have been proposed. Recurrent Neural Networks (RNN) [21] and Convolutional Neural Network (CNN) [22]–[24] models have achieved high levels of accuracy in state estimation. These models, however, perform state estimation independently without state transition information. Additionally, existing vision-based state estimation models have only extracted visual features from the entire endoscopic view [24]. The emphasis on regions in the endoscopic view that are more indicative of surgical states (e.g., areas surrounding the end-effectors), is helpful for eliminating environmental noises in visual features. This can be done through instrument tracking [25].

Deep learning-based methods for path and action prediction have been used in the field of computer vision, including path predictions using personal visual features and Long-Short Term Memory (LSTM) [26], [27] and early recognition of actions [28], [29]. But they have received little attention in surgical robotics. These methods have predicted human paths and actions seconds in advance. Liang et al. recently proposed a multi-task model for predicting a person's future path and activities in videos using various features, including the person's position, appearance, and interactions [30]. Compared to human activity datasets (such as ActEV [31]) which are used for human path and activity predictions, our problem has the privilege of having synchronized robot kinematics, endoscopic vision, and system events as data sources. This is especially useful in the prediction of surgical states, since different sources of input data have their respective strengths and weaknesses in representing states with various kinematics and visual features. Previously, we have proposed a unified model for surgical state estimation - Fusion-KVE - that incorporated multiple types of input data and exceeded the state-of-the-art state estimation performance [17]. Building on this, we explored the task of concurrent instrument path and surgical state predictions with multiple data streams and the incorporation of historic state transition sequences.

Contributions: This paper proposes *daVinciNet*, a joint prediction model of instrument paths and surgical states for RAS tasks. Our model uses multiple data types gathered from a da Vinci® surgical system as input (Fig. 1). The model performs feature extraction and makes multi-step predictions of both the end-effectors' trajectories in the endoscopic reference frame and the future surgical states. We aim for real-time predictions of up to 2 seconds in advance. Our main contributions include:

- Incorporating and extracting features from multiple available data sources, including robot kinematics, endoscopic vision, and system events;
- Implementing a vision-based tool tracking algorithm to determine the Regions of Interest (RoIs) of the endoscopic vision data for more localized and effective visual feature extraction;
- Applying a state estimation model (Fusion-KVE) to infer the historic surgical state sequence for state prediction;
- Achieving accurate trajectory and state predictions for up to 2 seconds by incorporating the temporal information in data sequences using learning-based methods.

Our model's performance was evaluated using the JIGSAWS suturing dataset [32] and the extended Robotic Intra-Operative Ultrasound (RIOUS+) imaging dataset [17]. The RIOUS+ dataset contains ultrasound imaging trials in various experimental settings (phantom, in-vivo, and cadaver) and endoscopic motion. We performed multi-step end-effector path and surgical state predictions for various time spans (0.1s to 2s). We also performed ablation studies [33] to better understand the contributions of various types of features used in both prediction tasks. To the best of our knowledge, there is no current benchmark for surgical instrument trajectory or state predictions. We show robustness of the *daVinciNet* by comparing it to its ablated versions. We hope that this paper and the future release of the RIOUS+ dataset will encourage future exploration of surgical scene prediction.

II. Method

We propose an end-to-end joint prediction model that concurrently predicts the end-effector trajectories and the surgical states, as shown in Fig. 2. Our model resembles the structure of a Long-Short Term Memory (LSTM) encoder-decoder model - a model widely used in natural language processing [34]. It consists of feature extraction components that take in the vision, kinematics, and events data sequences for a observation window with size T_{obs}. The outputs of the endoscopic vision module and the kinematics encoder are consolidated to a feature tensor $\boldsymbol{Q}$, which is fed to an attention-based LSTM model for multi-step trajectory prediction from time $t+1$ to T_{pred}, where T_{pred} is the number of prediction steps. The output of Fusion-KVE is an input for the surgical state prediction in addition to the feature tensor. In the following subsections, we will first describe the details in feature extraction and prediction modules, and then the training details of the model.

A. Visual Feature Encoder

We developed a novel endoscopic vision analysis module that extracts visual features at both global and local levels. The endoscopic scene features are extracted by a CNN-LSTM encoder model. We use a pre-trained VGG-16 model [35] to extract a fixed size CNN feature vector from each frame in the endoscope video. Instead of directly using the

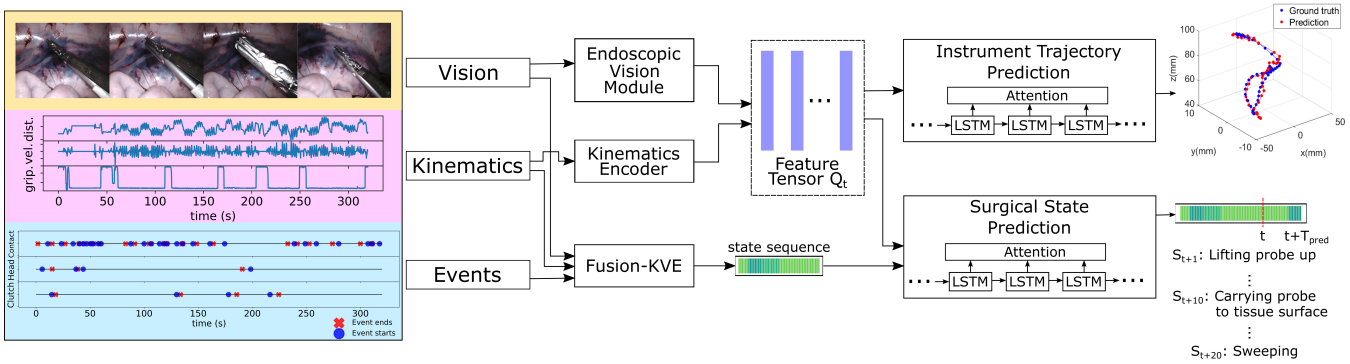

Fig. 2: *daVinciNet*'s model architecture. Given synchronized vision, robot kinematics and system events data streams, our model uses multiple encoders and Fusion-KVE to extract visual, kinematics, and states features. The concatenated feature tensor $\boldsymbol{Q}$ is used for both instrument trajectory and surgical state predictions. The state sequence, in addition to $\boldsymbol{Q}$, is the input of the surgical state prediction model. An attention-based LSTM decoder model was implemented for both prediction tasks. The example is shown with 10Hz data.

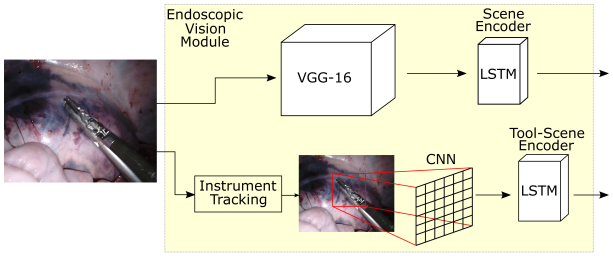

Fig. 3: Details of the endoscopic vision module, which extracts both global visual information from the entire endoscopic view and RoI visual information that focuses on the interactions between the end-effector and the surrounding scene. The RoIs are determined by a silhouette-based instrument tracking model [25].

temporal concatenation of CNN features as described in [22]–[24], we drew inspiration from vision-based human path prediction [30], [36] and implemented an LSTM encoder as the scene encoder (Fig. 3) to capture the the CNN feature series' long-term temporal dependencies. Given CNN features $\boldsymbol{X}_t^{scene} = (\boldsymbol{x}_{t-T_{obs}+1}^{scene}, ..., \boldsymbol{x}_t^{scene})$ with $\boldsymbol{x}_t^{scene} \in \mathbb{R}^m$, where m is the number of scene CNN features at each time-step, the LSTM encoder maps the hidden state $\boldsymbol{h}_t^{scene}$ from $\boldsymbol{x}_t^{scene}$ with:

$$\boldsymbol{h}_t^{scene} = LSTM(\boldsymbol{h}_{t-1}^{scene}, \boldsymbol{x}_t), \quad (1)$$

where $\boldsymbol{h}_t \in \mathbb{R}^{n_{scene}}$ and n_{scene} is the LSTM encoder's hidden size. The concatenated encoder hidden states $\boldsymbol{H}_t^{scene} = (\boldsymbol{h}_{t-T_{obs}+1}^{scene}, ..., \boldsymbol{h}_t^{scene}))$ forms a part of the feature tensor $\boldsymbol{Q}_t$.

The background environment in real-world RAS endoscopic vision is complex and varies significantly across cases. The effect of such environmental noise can be eliminated by using large annotated datasets with various backgrounds to train the aforementioned CNN-LSTM encoder; however, such datasets are expensive to acquire. It is worth noting that most states in surgical task FSMs are associated with the movements of the instrument end-effectors (Table I). We implemented a silhouette-based instrument tracking model following [25] and performed bounding box detection of the end-effectors from the endoscopic view. The surrounding areas (bounding box) of each instrument's coordinates from the endoscopic view are the RoIs, and were the input to a different CNN-LSTM encoder model for feature extraction (the tool-scene encoder in Fig. 3). The encoded RoI hidden state $\boldsymbol{H}_t^{RoI} \in \mathbb{R}^{T_{obs} \times n_{RoI}}$ is also a part of $\boldsymbol{Q}_t$.

Implementation details: Each endoscope video frame was resized to a $224 \times 224 \times 3$ RGB image before being input to the VGG-16 model. The VGG-16 model was pre-trained following our previous work [17]. $m = 1024$ CNN features were extracted. The original $1280 \times 1024 \times 3$ RGB image of each frame was then input to the instrument tracking model. If there are instruments in the endoscopic view, the RoIs around the instruments' end-effectors were extracted for 100 CNN features through two layers of CNNs with ReLU [37] activation. The scene encoder and the tool-scene encoder both have $n_{scene} = n_{RoI} = 32$ hidden states.

B. Attention-based Kinematics Feature Encoder

To extract kinematics features from multiple da Vinci® surgical system data inputs (end-effectors' translational and rotational positions, etc.), and capture the long-term data progress, we followed [38] and implemented an LSTM encoder with input attention to identify the importance of different driving series. At time t, the kinematics input is $\boldsymbol{X}_t^{kin} = (\boldsymbol{x}_{t-T_{obs}+1}^{kin}, ..., \boldsymbol{x}_t^{kin})$, where $\boldsymbol{x}_t^{kin} \in \mathbb{R}^l$ and l is the number of kinematics input series. Instead of deriving $\boldsymbol{h}_t^{kin}$ directly from Eq. (1), we constructed the input attention mechanism by learning a multiplier vector that represents the weights of each input series at time t from the previous hidden state $\boldsymbol{h}_{t-1}^{kin}$ and the LSTM unit's cell state $\boldsymbol{s}_{t-1}^{kin}$:

$$\boldsymbol{\alpha}_t^i = softmax(tanh(\boldsymbol{W}_e(\boldsymbol{h}_{t-1}^{kin}, \boldsymbol{s}_{t-1}^{kin}) + \boldsymbol{V}_e \boldsymbol{x}^{kin,i})), \quad (2)$$

where $\boldsymbol{x}^{kin,i} \in \mathbb{R}^{T_{obs}}$ is the i-th kinematics input series ($1 \leq i \leq l$), and $\boldsymbol{W}_e$ and $\boldsymbol{V}_e$ are learnable encoder parameters. A

softmax function normalizes the attention weights $\boldsymbol{\alpha}_t$. The weighted kinematics input at time t is then:

$$\tilde{\boldsymbol{x}}_t^{kin} = \sum_{i=1}^{m} \alpha_t^i \boldsymbol{x}_t^{kin,i}, \tag{3}$$

which substitutes $\boldsymbol{x}_t$ in Eq. (1). The encoded hidden states $\boldsymbol{H}_t^{kin} = (\boldsymbol{h}_{t-T_{obs}+1}^{kin}, ..., \boldsymbol{h}_t^{kin}) \in \mathbb{R}^{T_{obs} \times n_{kin}}$ is the final component of feature tensor $\boldsymbol{Q}_t$.

Implementation details: the JIGSAWS suturing dataset and the RIOUS+ dataset respectively contain 26 and 16 kinematic variables. Except for the target variables (the end-effectors' paths), we used $l = 20, 13$ input series for the JIGSAWS and the RIOUS+ dataset, respectively. The kinematics encoder had $n_{kin} = 32$ hidden states.

C. Instrument Path and Surgical State Predictions

After encoding, a feature tensor $\boldsymbol{Q} = (\boldsymbol{q}_{t-T_{obs}+1}, ..., \boldsymbol{q}_t) \in \mathbb{R}^{T_{obs} \times (n_{scene}+n_{RoI}+n_{kin})}$ was obtained. We implemented LSTM decoders to predict the Cartesian instrument paths in the endoscopic reference frame after time t ($\boldsymbol{y}_t \in \mathbb{R}^{T_{pred} \times r}$) and future states ($s_t \in \mathbb{R}^{T_{pred}}$), respectively. T_{pred} is the number of prediction time-steps. The LSTM decoders were implemented with temporal attention [39] to alleviate the performance deterioration as the input sequences' lengths T_{obs} increase [34]. The temporal attention mechanism allows the decoders to use relevant hidden states among all time-steps from $\boldsymbol{Q}$ in an adaptive manner. At time t, the temporal attention weights $\boldsymbol{\beta} \in \mathbb{R}^{T_{obs}}$ of the decoder hidden state $\boldsymbol{d}_t \in \mathbb{R}^{n'}$ is learned from $\boldsymbol{d}_{t-1}$, the previous cell state of the decoder LSTM unit $\boldsymbol{c}_{t-1}$, and the feature tensor:

$$\boldsymbol{\beta}_t^j = softmax(tanh(\boldsymbol{W}_d(\boldsymbol{d}_{t-1}, \boldsymbol{c}_{t-1}) + \boldsymbol{V}_d \boldsymbol{q}_j)), \tag{4}$$

where n' is the hidden size, $\boldsymbol{W}_d$ and $\boldsymbol{V}_d$ are decoder parameters to be learned.

The weighted feature $\tilde{\boldsymbol{q}}_t = \sum_{j=1}^{T_{obs}} \beta_t^j \boldsymbol{q}_t^j$ and the historic target sequences (3-D end-effector path $\boldsymbol{y}$ or estimated surgical state s) from $t - T_{obs} + 1$ to t were used to extract the target embedding following [30]:

$$\tilde{\boldsymbol{y}}_{t-1} = \boldsymbol{W}_{targ}(\tilde{\boldsymbol{q}}_{t-1}, \boldsymbol{y}_{t-1}) + \boldsymbol{V}_{targ}, \tag{5}$$

where $\boldsymbol{W}_{targ}$ and $\boldsymbol{V}_{targ}$ are learned. The update of $\boldsymbol{d}_t$ is:

$$\boldsymbol{d}_t = LSTM(\boldsymbol{d}_{t-1}, [\tilde{\boldsymbol{y}}_{t-1}, \tilde{\boldsymbol{q}}_t]), \tag{6}$$

after which the end-effector trajectory predictions $\hat{\boldsymbol{y}}_t$ are computed by a fully connected layer. The probability vector $\boldsymbol{s}$ for surgical state prediction can be similarly derived. The state prediction $\hat{s}_t \in \mathbb{R}^{T_{pred}}$ is the future state sequence, with each state having the maximum likelihood among all states at each time-step.

It is worth noting that in order to obtain the historic sequence of surgical state s from $t - T_{obs} + 1$ to t, we implemented Fusion-KVE - a unified surgical state estimation model we proposed recently [17]- instead of using the ground truth (GT) state sequence. In real-time RAS settings, the surgical state prediction model does not have access to the manually-labeled historic surgical state sequence; therefore, a state estimation model is needed to provide the historic state sequence. For the JIGSAWS suturing dataset, we implemented the ablated version of the state estimation model (Fusion-KV) due to the lack of system events data. Section IV will discuss the performance difference when various features are included for both prediction tasks.

Implementation details: We implemented both trajectory and state prediction LSTM decoders with $n' = 96$ hidden states after grid search for parameters. $r = 6$ variables (3D end-effector paths for both instruments) were predicted for the JIGSAWS data set, while $r = 3$ variables were predicted for the RIOUS+ dataset. Multi-step predictions were implemented, with $T_{obs} = 20$ and $max(T_{pred}) = 20$ for data streaming at 10Hz.

D. Training

The entire model, including the feature extraction and the prediction modules, was trained end-to-end with the goal of minimizing a loss function that accounts for both the trajectory prediction and state prediction accuracies. The trajectory loss function is the cumulative L_2 loss between the predicted end-effector trajectory and the GT trajectory, summed up from $T_{obs}+1$ to T_{pred}. The state estimation loss function is the cumulative categorical cross-entropy loss that accounts for the discrepancies between the predicted surgical states and the GT.

III. Experimental Evaluations

We evaluated our trajectory and state prediction models on the JIGSAWS and RIOUS+ data sets (see Table I).

A. Datasets

JIGSAWS: The JIGSAWS dataset includes three types of RAS tasks performed in a benchtop setting [9]. Each trial lasts around 1.5 minutes and contains synchronized endoscopic video and robot kinematics data. We used the 39 suturing trials for model evaluation, which has 9 possible actions (Table I). The kinematics data series included the

TABLE I: Datasets State Descriptions and Duration

JIGSAWS Suturing Dataset

Action ID	Description	Duration (s)
G1	Reaching for the needle with right hand	2.2
G2	Positioning the tip of the needle	3.4
G3	Pushing needle through the tissue	9.0
G4	Transferring needle from left to right	4.5
G5	Moving to center with needle in grip	3.0
G6	Pulling suture with left hand	4.8
G7	Orienting needle	7.7
G8	Using right hand to help tighten suture	3.1
G9	Dropping suture and moving to end points	7.3

RIOUS+ Dataset

State ID	Description	Duration (s)
S1	Probe released, out of endoscopic view	6.3
S2	Probe released, in endoscopic view	7.6
S3	Reaching for probe	3.1
S4	Grasping probe	1.1
S5	Lifting probe up	2.4
S6	Carrying probe to tissue surface	2.3
S7	Sweeping	5.1
S8	Releasing probe	1.7

end-effectors' positions, velocities, and gripper angles of the universal patient-side manipulators (USM). We converted the rotation matrices that represent the two end-effectors' orientations into Euler angles to reduce data dimensionality.

RIOUS+: The RIOUS+ dataset is an extended version of RIOUS, which was first introduced in [17], where the dataset was used to evaluate surgical state estimation accuracy. The surgical task performed is ultrasound scanning, which is a commonly used da Vinci intra-operative procedure to understand a patient's anatomic structures. Comparing to JIGSAWS, the RIOUS+ dataset contains more real-world RAS elements. RIOUS+ contains 40 trials performed by 5 users, 27 of which were performed on a phantom kidney in a dry-lab setting, 9 were on a porcine kidney, and 4 were on a cadaver liver performed in operating room settings. For each trial, RIOUS+ includes synchronized endoscopic vision, robot kinematics, and system events data collected from a da Vinci® Xi surgical system. The position, velocity, and gripper angles of the USM were included as well as the endoscope position. The system events data are represented as binary time series, including surgeon head in/out, two ultrasound probe events, master clutch, camera follow, and instrument follow. The ultrasound imaging task has 8 possible states (Table I).

B. Metrics

We use three metrics to evaluate the accuracy of our end-effector trajectory prediction: Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) [38], [40]:

$$\begin{aligned} RMSE &= \sqrt{\frac{\sum_{i=1}^{N}(y^i - \hat{y}^i)^2}{N}} \\ MAE &= \frac{\sum_{i=1}^{N}|y^i - \hat{y}^i|}{N} \\ MAPE &= \sum_{i=1}^{N}\left|\frac{y^i - \hat{y}^i}{y^i}\right| \times 100\% \end{aligned} \tag{7}$$

Since RMSE and MAE are independent of the variables' absolute values, they provide an intuitive comparison among variables in the same datset. MAPE calculates the percentage error; therefore, it provides a direct comparison between prediction accuracies across different datasets. To evaluate the surgical state prediction accuracy, we calculated the percentage of accurately predicted frames in the testing sequences [17], [24]. We evaluate both prediction tasks and both datasets using *Leave One User Out* as described in [32].

Both trajectory and state predictions are performed in a multi-step manner for up to 2-second into the future ($max(T_{pred}) = 20$ for 10Hz data streams). The model performances with respect to the number of prediction time-steps are discussed in the next section. For each prediction time-step, the evaluation metrics are based only on the prediction at that time-step, without accounting for the previous prediction steps. To describe the representative performance of our model, we select a 1 second prediction time-step for Fig. 6, Table II, and Table III.

IV. Results and Discussions

Tables II and III summarize *daVinciNet*'s end-effector path prediction and surgical state prediction performance on the JIGSAWS suturing and RIOUS+ datasets when the prediction time-step is 1 second ($T_{pred} = 10$). Fig. 4 and Fig. 5 illustrate how our model performance changes at various prediction time-steps. Both tables and figures also include the performance of ablated versions of our model as compared to their performances on the full model. Fig. 6 shows a sample sequence of ultrasound imaging task state prediction when $T_{pred} = 10$ using our state prediction model as well as ablated versions of it.

Table II compares the differences in end-effector path prediction accuracy when *daVinciNet* uses only kinematics features $\boldsymbol{H}^{kin}$, scene and kinematics features $\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$, and a full feature tensor $\boldsymbol{Q} = \{\boldsymbol{H}^{scene}, \boldsymbol{H}^{RoI}, \boldsymbol{H}^{kin}\}$. The accuracy of end-effector trajectory prediction in the endoscopic reference frame was evaluated, along with the end-effector distance $d = \sqrt{x^2 + y^2 + z^2}$ from the origin (camera tip). *daVinciNet* predictions based on all data streams consistently achieve up to 20% better performance. Clearly, vision features contribute to better prediction of the end-effector trajectory in the endoscopic reference frame. Many instrument movements have advanced visual cues. E.g., suture pulling usually occurs after the needle tip has appeared on the suturing pad or tissue. Visual features, such as the

JIGSAWS Suturing

		x_1	y_1	z_1	d_1	x_2	y_2	z_2	d_2
$\boldsymbol{H}^{kin}$	RMSE	2.81	2.42	3.28	4.16	3.8	4.26	4.75	5.92
	MAE	2.19	1.95	2.86	3.7	3.42	3.91	4.31	5.34
	MAPE	6.8	6.09	7.39	8.93	7.77	8.03	8.2	10.14
$\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$	RMSE	2.7	2.29	3.25	4.01	3.65	4.01	4.63	5.2
	MAE	2.17	1.88	2.79	3.5	3.15	3.7	4.16	4.76
	MAPE	6.73	5.88	7.18	8.44	7.05	7.5	7.91	9.27
$\boldsymbol{Q}$	RMSE	2.53	1.89	2.96	3.35	3.15	3.5	3.91	4.51
	MAE	2.07	1.51	2.46	3.09	2.78	3.06	3.5	4.17
	MAPE	6.43	4.72	6.35	7.46	6.13	6.11	6.67	7.95

RIOUS+

		x	y	z	d
$\boldsymbol{H}^{kin}$	RMSE	1.67	1.8	1.22	2.3
	MAE	1.45	1.62	1.24	2.06
	MAPE	1.89	2.62	1.76	2.17
$\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$	RMSE	1.67	1.7	1.18	2.1
	MAE	1.33	1.52	1.12	1.91
	MAPE	1.7	2.43	1.57	2.01
$\boldsymbol{Q}$	RMSE	1.23	1.41	1.08	1.98
	MAE	1.09	1.34	0.97	1.64
	MAPE	1.31	2.16	1.1	1.72

Table II: End-effector trajectory prediction performance measures when predicting one second ahead ($T_{pred} = 10$). The prediction performances for the Cartesian end-effector path in the endoscopic reference frame (x, y, z) and $d = \sqrt{x^2 + y^2 + z^2}$ are compared when the trajectory prediction decoder uses only kinematics features ($\boldsymbol{H}^{kin}$), uses global scene and kinematics features ($\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$), and uses global scene, RoI, and kinematics features ($\boldsymbol{Q}$).

Input data	JIGSAWS Suturing (%)	RIOUS+ (%)
Q only	64.11	65.44
Fusion-KVE only	75.08	76.5
Q+Fusion-KVE	**84.3**	**91.02**

Table III: Surgical State Prediction performance when predicting one second ahead ($T_{pred} = 10$). The prediction performances are compared when the state prediction decoder uses only the feature tensor ($\boldsymbol{Q}$ only), only the historic state sequence (Fusion-KVE only), and both ($\boldsymbol{Q}$+Fusion-KVE).

distance between the end-effector and nearby tissue, also help in predicting trajectory changes. Therefore, including visual features, especially RoI information around the end-effectors, is helpful in end-effector trajectory prediction. Table III investigates the surgical state prediction accuracy with only $\boldsymbol{Q}$, only state estimation results, or both as input features. The significant improvement in state prediction accuracy by incorporating both data sources supports our model design. As mentioned in the previous section, *daVinciNet* does not have access to ground truth state sequence in real-time prediction. The high prediction accuracy using an estimated state sequence shows the robustness of our model in real-time state prediction.

Fig. 4 shows how end-effector trajectory prediction accuracy changes with increasing prediction time-step. We compared trajectory prediction MAE when the feature tensor includes only $\boldsymbol{H}^{kin}$, $\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$, or all three types of features, $\boldsymbol{Q}$. For JIGSAWS, the MAE of the left (d_1) and right (d_2) instruments are averaged. The use of RoI visual features consistently decreases the trajectory prediction MAE, especially at large prediction steps. This observation reaffirms our discussion earlier that the visual features and advanced cues concentrated in RoIs were detected by the tool-scene encoder and contributed to end-effector trajectory prediction.

Fig. 5 shows the progress of surgical state prediction accuracy as prediction time-step increases. For surgical state prediction, we compared performance when the state prediction decoder is based only on $\boldsymbol{Q}$, only on the Fusion-KVE state sequence results, and both. The ablated prediction models were constructed with the $\tilde{\boldsymbol{q}}$ or $\boldsymbol{y}$ term omitted from Eq. (4)-(6), respectively. When only feature tensor $\boldsymbol{Q}$ was available to the state prediction decoder, state prediction accuracy is constantly the lowest and exponentially decreases as the prediction time-step increase. Due to the absence of historic target series, the state prediction model can only rely on predicted state sequences from previous time-steps, which causes previous prediction errors to affect the next state prediction with little correction. When the state prediction decoder can use historic state estimation sequences from Fusion-KVE, prediction accuracy is improved, especially for large prediction steps, since the target series provide corrections to previous prediction errors. When the decoder receives only the target series with no additional feature series, the surgical state predictions are performed without visual or kinematic cues that indicate state changes; therefore, the improvement in prediction accuracy is limited. By incorporating both $\boldsymbol{Q}$ and Fusion-KVE, advanced cues from visual and kinematics features are used to forecast state changes, and the historic state sequence provides corrections to prediction errors. *daVinciNet* achieves the highest state prediction accuracy that does not show significant deterioration as prediction time-step increases.

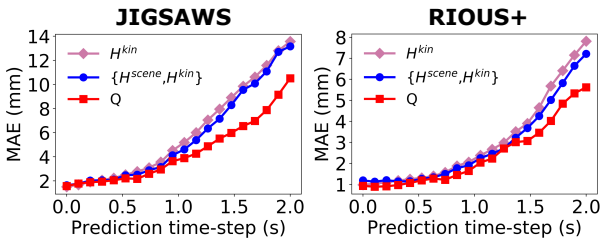

Fig. 4: Model performance comparisons when different features are included for end-effector trajectory prediction for various prediction time-steps. The model was constructed with only kinematics features ($\boldsymbol{H}^{kin}$), scene and kinematics features ($\{\boldsymbol{H}^{scene}, \boldsymbol{H}^{kin}\}$), and scene, RoI and kinematics features ($\boldsymbol{Q}$). $mean(MAE_{d_1} + MAE_{d_2})$ and MAE_d were plotted for JIGSAWS and RIOUS+, respectively.

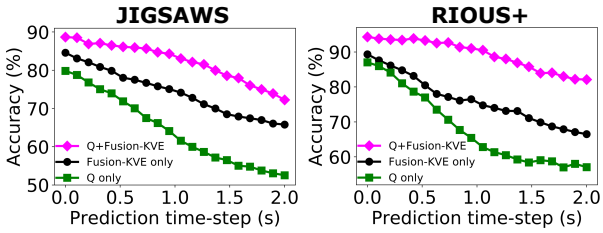

Fig. 5: Model performance comparisons when different features are included for surgical state prediction. The state prediction decoder was trained with only feature tensor ($\boldsymbol{Q}$ only), only historic state sequence (Fusion-KVE only), and both ($\boldsymbol{Q}$+Fusion-KVE).

The sample sequence of ultrasound imaging state prediction results in Fig. 6 further supports our model architecture's inclusion of the feature tensor $\boldsymbol{Q}$ and Fusion-KVE output for surgical state prediction. Prediction errors occur in blocks when only $\boldsymbol{Q}$ is used for prediction, due to uncorrected errors from using the predicted state sequence from previous time-steps. A model using only Fusion-KVE shows fewer errors in consecutive time blocks; however, the missing feature input leads to delayed responses to real world state changes, or even missing states with relatively shorter duration. A *daVinciNet* model that incorporates both $\boldsymbol{Q}$ and Fusion-KVE significantly improves state prediction accuracy, with the remaining errors located mostly around state transitions.

Additionally, since the temporal annotations of surgical states were done manually by humans, we investigated the annotation variance in the ground truth state sequence introduced by human annotators. Five users were asked to annotate the sample sequence in Fig. 6 frame-by-frame with states in Table I. The discrepancies among annotations are

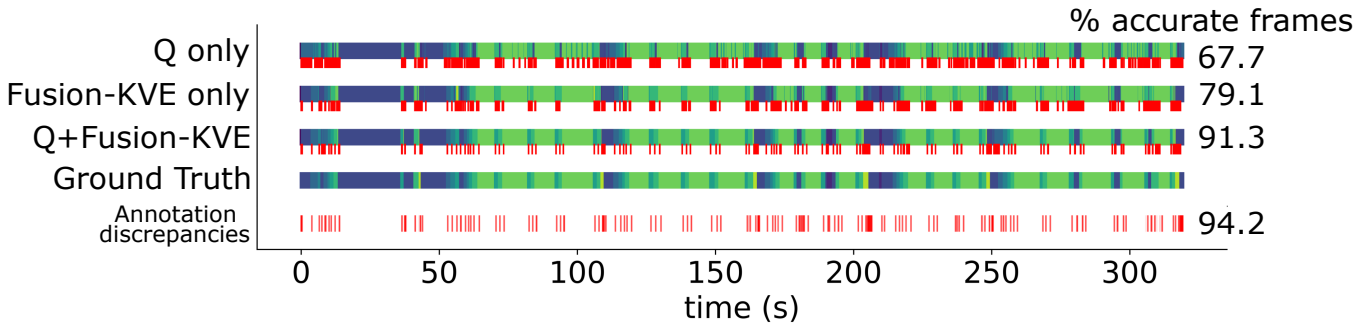

Fig. 6: Sample 1-second ultrasound imaging state prediction results using only the feature tensor (**Q** only), only the historic state sequence (Fusion-KVE only), and both (**Q**+Fusion-KVE). Each block bar contains the state prediction results when $T_{pred} = 10$ (top), and the discrepancies between the prediction results and the GT shown in red. The Annotation discrepancies row (bottom) shows the locations of frames where multiple annotators used different state labels, with the mean matching rate of 94.2% among annotators.

shown in the bottom row of Fig. 6, with an average matching rate of 94.2% among annotators. Even human annotators cannot agree perfectly on a state sequence: their discrepancies occur mostly at state transitions, which is expected since the transition from one state to another in a surgical subtask is not abrupt, but gradual. Hence, annotators may identify different video frames as state transitions. Hence, *daVinciNet* state prediction errors can be partially attributed to human annotation errors in identifying the exact state transition times. Thus, *daVinciNet*'s robustness is further established by its high state prediction accuracy even in the presence of noise in the ground truth data.

V. Conclusions and Future Work

This paper focused on real-time prediction of variables that are crucial to RAS, including instrument end-effector trajectories in endoscopic viewing frames and discrete surgical states. We proposed the *daVinciNet*: a unified end-to-end joint prediction model that uses synchronized sequences of robot kinematics, endoscopic vision, and system events data as input to predict instrument trajectories and surgical states. Our model achieves accurate predictions of the end-effector path, with distance error as low as 1.64mm and MAPE of 1.72% when predicting the end-effector location 1-second in the future. The surgical state estimation accuracy achieved by the *daVinciNet* is up to 91.02%, and compares well with human annotator accuracy of 94.2%. By accurately predicting the end-effector trajectory and surgical states in datasets with various experimental settings and robot motions, the *daVinciNet* proves its robustness in realistic RAS tasks.

We further illustrated the necessity and advantages of including multiple data sources for joint prediction tasks by comparing the performance of our full model against ablated versions. Improved performance arises, for instance, because many instrument movements have visual features and advanced cues that can be captured by *daVinciNet*'s endoscopic vision feature module. Including a full feature tensor with the historic state sequence also significantly improves accuracy in surgical state prediction, compared to only using one type of input. Richer information regarding surgical subtasks can be extracted from multiple encoders, which leads to more accurate predictions. We also showed the sizeable contribution of RoI visual features to performance. *daVinciNet* incorporates a silhouette-based instrument tracking algorithm to identify the RoIs in endoscopic vision and our Fusion-KVE state estimation model [17] to obtain the historic surgical state sequence. The applications of existing surgical scene-understanding models allow us to achieve better performances in prediction.

To further improve prediction performance, a possible next step would be to incorporate semantic segmentation of endoscopic images. While extracting both global and RoI visual features provides *daVinciNet* with rich information and advanced cues for better predictions, the backgrounds in real-world RAS video is complicated and diverse. A semantic segmentation model maps each pixel of the endoscopic vision to a semantic class and therefore reduces the environmental noise. We also plan to apply the *daVinciNet* to RAS applications such as multi-agent systems with shared control or supervised autonomous surgical subtasks.

Acknowledgment

This work was funded by Intuitive Surgical, Inc. We would like to thank Dr. Azad Shademan and Dr. A. Jonathan McLeod for their support of this research.

References

[1] M. Selvaggio, G. A. Fontanelli, F. Ficuciello, L. Villani, and B. Siciliano, "Passive virtual fixtures adaptation in minimally invasive robotic surgery," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3129–3136, 2018.

[2] A. Shademan, R. S. Decker, J. D. Opfermann, S. Leonard, A. Krieger, and P. C. Kim, "Supervised autonomous robotic soft tissue surgery," *Sci. Rrans. Med.*, vol. 8, no. 337, pp. 337ra64–337ra64, 2016.

[3] S. A. Pedram, P. Ferguson, J. Ma, E. Dutson, and J. Rosen, "Autonomous suturing via surgical robot: An algorithm for optimal selection of needle diameter, shape, and path," in *IEEE Int. Conf. Robotics and Automation*, 2017, pp. 2391–2398.

[4] P. Chalasani, A. Deguet, P. Kazanzides, and R. H. Taylor, "A computational framework for complementary situational awareness (csa) in surgical assistant robots," in *IEEE Int. Conf. Robotic Computing*, 2018, pp. 9–16.

[5] S. P. DiMaio, C. J. Hasser, R. H. Taylor, D. Q. Larkin, P. Kazanzides, A. Deguet, B. P. Vagvolgyi, and J. Leven, "Interactive user interfaces for minimally invasive telesurgical systems," Feb. 15 2018, uS Patent App. 15/725,271.

[6] P. C. Kim, A. Krieger, Y. Kim, A. Shademan, and S. Leonard, "Automated surgical and interventional procedures," Dec. 29 2015, uS Patent 9,220,570.

[7] G.-Z. Yang, J. Cambias, K. Cleary, E. Daimler, J. Drake, P. E. Dupont, N. Hata, P. Kazanzides, S. Martel, R. V. Patel, *et al.*, "Medical robotics—regulatory, ethical, and legal considerations for increasing levels of autonomy," *Science Robotics*, vol. 2, no. 4, p. 8638, 2017.

[8] O. Weede, H. Mönnich, B. Müller, and H. Wörn, "An intelligent and autonomous endoscopic guidance system for minimally invasive surgery," in *IEEE Int. Conf. Robotics and Automation*, 2011, pp. 5762–5768.

[9] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. B. Haro, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager, "A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 9, pp. 2025–2041, 2017.

[10] A. Zia, C. Zhang, X. Xiong, and A. M. Jarc, "Temporal clustering of surgical activities in robot-assisted surgery," *Int. J. Computer Assisted Radiology and Surgery*, vol. 12, no. 7, pp. 1171–1178, 2017.

[11] N. Padoy, "Machine and deep learning for workflow recognition during surgery," *Minimally Invasive Therapy & Allied Technologies*, vol. 28, no. 2, pp. 82–90, 2019.

[12] C. Staub, S. Can, B. Jensen, A. Knoll, and S. Kohlbecher, "Human-computer interfaces for interaction with surgical tools in robotic surgery," in *IEEE RAS & EMBS Int. Conf. Biomed. Robotics and Biomechatronics*, 2012, pp. 81–86.

[13] C. Staub, C. Lenz, G. Panin, A. Knoll, and R. Bauernschmitt, "Contour-based surgical instrument tracking supported by kinematic prediction," in *IEEE RAS & EMBS Int. Conf. Biomed. Robotics and Biomechatronics*, 2010, pp. 746–752.

[14] W. Zhao, C. J. Hasser, W. C. Nowlin, and B. D. Hoffman, "Methods and systems for robotic instrument tool tracking with adaptive fusion of kinematics information and image information," Jan. 31 2012, uS Patent 8,108,072.

[15] Z. Wang, B. Zi, H. Ding, W. You, and L. Yu, "Hybrid grey prediction model-based autotracking algorithm for the laparoscopic visual window of surgical robot," *Mechanism and Machine Theory*, vol. 123, pp. 107–123, 2018.

[16] Y. Sun, B. Pan, Y. Fu, and G. Niu, "Visual based autonomous field of view control of laparoscope with safety-rcm constraints for semi-autonomous surgery," *Int. J. Medical Robotics and Computer Assisted Surgery*, p. e2079, 2020.

[17] Y. Qin, S. A. Pedram, S. Feyzabadi, M. Allan, A. J. McLeod, J. W. Burdick, and M. Azizian, "Temporal segmentation of surgical sub-tasks through deep learning with multiple data sources," *arXiv preprint arXiv:2002.02921*, 2020.

[18] L. Tao, E. Elhamifar, S. Khudanpur, G. D. Hager, and R. Vidal, "Sparse hidden markov models for surgical gesture classification and skill evaluation," in *Int. conf. information processing in computer-assisted interventions*. Springer, 2012, pp. 167–177.

[19] J. Rosen, J. D. Brown, L. Chang, M. N. Sinanan, and B. Hannaford, "Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model," *IEEE Trans. Biomedical engineering*, vol. 53, no. 3, pp. 399–413, 2006.

[20] M. Volkov, D. A. Hashimoto, G. Rosman, O. R. Meireles, and D. Rus, "Machine learning and coresets for automated real-time video segmentation of laparoscopic and robot-assisted surgery," in *IEEE Int. Conf. Robotics and Automation*. IEEE, 2017, pp. 754–759.

[21] R. DiPietro, C. Lea, A. Malpani, N. Ahmidi, S. S. Vedula, G. I. Lee, M. R. Lee, and G. D. Hager, "Recognizing surgical activities with recurrent neural networks," in *Int. Conf. medical image computing and computer-assisted intervention*. Springer, 2016, pp. 551–558.

[22] T. Yu, D. Mutter, J. Marescaux, and N. Padoy, "Learning from a tiny dataset of manual annotations: a teacher/student approach for surgical phase recognition," *arXiv preprint arXiv:1812.00033*, 2018.

[23] C. Lea, A. Reiter, R. Vidal, and G. D. Hager, "Segmental spatiotemporal cnns for fine-grained action segmentation," in *European Conf. Computer Vision*. Springer, 2016, pp. 36–52.

[24] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *European Conf. Computer Vision*. Springer, 2016, pp. 47–54.

[25] M. Allan, S. Ourselin, D. J. Hawkes, J. D. Kelly, and D. Stoyanov, "3-d pose estimation of articulated instruments in robotic minimally invasive surgery," *IEEE trans. medical imaging*, vol. 37, no. 5, pp. 1204–1213, 2018.

[26] J. F. P. Kooij, N. Schneider, F. Flohr, and D. M. Gavrila, "Context-based pedestrian path prediction," in *European Conf. Computer Vision*. Springer, 2014, pp. 618–633.

[27] T. Yagi, K. Mangalam, R. Yonetani, and Y. Sato, "Future person localization in first-person videos," in *IEEE Conf. Computer Vision and Pattern Recog.*, 2018, pp. 7593–7602.

[28] M. Sadegh Aliakbarian, F. Sadat Saleh, M. Salzmann, B. Fernando, L. Petersson, and L. Andersson, "Encouraging lstms to anticipate actions very early," in *IEEE Int. Conf. Computer Vision*, 2017, pp. 280–289.

[29] S. Ma, L. Sigal, and S. Sclaroff, "Learning activity progression in lstms for activity detection and early detection," in *IEEE Conf. Computer Vision and Pattern Recog.*, 2016, pp. 1942–1950.

[30] J. Liang, L. Jiang, J. C. Niebles, A. G. Hauptmann, and L. Fei-Fei, "Peeking into the future: Predicting future person activities and locations in videos," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2019, pp. 5725–5734.

[31] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, D. Joy, A. Delgado, A. Smeaton, Y. Graham, *et al.*, "Trecvid 2018: Benchmarking video activity detection, video captioning and matching, video storytelling linking and video search," 2018.

[32] Y. Gao, S. S. Vedula, C. E. Reiley, N. Ahmidi, B. Varadarajan, H. C. Lin, L. Tao, L. Zappella, B. Béjar, D. D. Yuh, *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling," in *MICCAI Workshop: M2CAI*, vol. 3, 2014, p. 3.

[33] R. Meyes, M. Lu, C. W. de Puiseau, and T. Meisen, "Ablation studies in artificial neural networks," *arXiv preprint arXiv:1901.08644*, 2019.

[34] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[36] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *IEEE Conf. computer Vision and Pattern Recog.*, 2016, pp. 961–971.

[37] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Int. Conf. Machine Learning*, 2010, pp. 807–814.

[38] Y. Qin, D. Song, H. Chen, W. Cheng, G. Jiang, and G. Cottrell, "A dual-stage attention-based recurrent neural network for time series prediction," *arXiv preprint arXiv:1704.02971*, 2017.

[39] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[40] M. Plutowski, G. Cottrell, and H. White, "Experience with selecting exemplars from clean data," *Neural Networks*, vol. 9, no. 2, pp. 273–294, 1996.