Estimation of Interaction Forces in Robotic Surgery using a Semi-Supervised Deep Neural Network Model

Arturo Marban[1,2] Vignesh Srinivasan[2], Wojciech Samek[2], *Member, IEEE,*
Josep Fernández[1], Alicia Casals[1], *Senior Member, IEEE,*

Abstract— **Providing force feedback as a feature in current Robot-Assisted Minimally Invasive Surgery systems still remains a challenge. In recent years, Vision-Based Force Sensing (VBFS) has emerged as a promising approach to address this problem. Existing methods have been developed in a Supervised Learning (SL) setting. Nonetheless, most of the video sequences related to robotic surgery are not provided with ground-truth force data, which can be easily acquired in a controlled environment. A powerful approach to process unlabeled video sequences and find a compact representation for each video frame relies on using an Unsupervised Learning (UL) method. Afterward, a model trained in an SL setting can take advantage of the available ground-truth force data. In the present work, UL and SL techniques are used to investigate a model in a Semi-Supervised Learning (SSL) framework, consisting of an encoder network and a Long-Short Term Memory (LSTM) network. First, a Convolutional Auto-Encoder (CAE) is trained to learn a compact representation for each RGB frame in a video sequence. To facilitate the reconstruction of high and low frequencies found in images, this CAE is optimized using an adversarial framework and a $L1$-loss, respectively. Thereafter, the encoder network of the CAE is serially connected with an LSTM network and trained jointly to minimize the difference between ground-truth and estimated force data. Datasets addressing the force estimation task are scarce. Therefore, the experiments have been validated in a custom dataset. The results suggest that the proposed approach is promising.**

Index Terms— **Vision Based Force Sensing, Robotic Surgery, Deep Neural Networks, Semi-Supervised Learning.**

I. INTRODUCTION

Force feedback is a desired feature in Robot-Assisted Minimally Invasive Surgery systems. It allows the integration of the "sense of touch", resulting in potential benefits. For instance, improved manipulation of human soft-tissues avoiding damage due to excessive applied forces. However, the integration of force feedback in surgical robotic systems still remains an open problem [1][2]. Advances in the fields of computer vision and artificial intelligence have resulted in an emerging research area known as Vision Based Force Sensing (VBFS). VBFS enables the estimation of interaction forces between surgical instruments and soft-tissue by processing video sequences. Such data is easily provided by surgical robotic systems, nonetheless, its interpretation is challenging.

Different methods have been proposed to address VBFS in robotic-assisted surgery scenarios. They estimate forces from (monocular/stereo) video sequences relying on an accurate modeling of soft-tissues' deformation (in 3D space) caused by the interaction with surgical instruments. Moreover, in VBFS, the processing of the surgical tool motion is beneficial (i.e. the tool-tip trajectory). A VBFS approach was investigated in [3] using a simplified scenario consisting of a rubber membrane. Its deformation was recovered by tracking nodal displacements and a finite element method was used to model the mechanical relationship between deformation and force. A more realistic scenario was studied in [4], which addresses monocular force estimation using a real lamb liver as experimental material. The authors proposed a virtual template to model soft-tissue surface deformation. However, it is assumed that the soft-tissue surface behaves as a smooth function with local deformation. The relationship between force and penetration depth caused by the surgical tool was modeled based on a stress-strain bio-mechanical model. VBFS applied to neurosurgery was investigated in [5] and [6]. In [5], soft-tissue surface deformation is computed using a depth map extracted from stereo-endoscopic images. Then, a surface mesh based on spring-damper models processes this information to render force as output. In contrast, the authors in [6] developed a method based on quasi-dense stereo correspondence to recover surface deformation from stereo video sequences. Afterward, force is estimated from the surgical tool displacement (which is extracted from the deformation data), using a 2nd order polynomial model. Models based on neural networks have been investigated in recent years. For instance, [7] proposed a 3D lattice in a minimization framework for modeling the complex deformation of soft-tissues. Furthermore, a recurrent neural network was designed to estimate force by processing the information provided by this lattice in addition to the surgical tool motion. Subsequent notable works by the same author include [8] and [9], in which the recurrent neural network described in [7] is improved by designing a model based on the Long-Short Term Memory (LSTM) network architecture [10], achieving high accuracy in the estimation of forces (in 3D space).

The literature review of VBFS in robotic-assisted surgery, reveals that the proposed neural network models have been designed in a Supervised Learning (SL) setting. However, the advantages of using a Semi-Supervised Learning (SSL)

[1] Josep Fernández and Alicia Casals, are with the Research Centre for Biomedical Engineering (CREB), Universitat Politècnica de Catalunya, 08034, Barcelona, Spain e-mail: josep.fernandez@upc.edu, alicia.casals@upc.edu

[2] Arturo Marban, Vignesh Srinivasan and Wojciech Samek, are with Fraunhofer Heinrich Hertz Institute, Einsteinufer 37, 10587 Berlin, Germany e-mail: arturo.marban@hhi-extern.fraunhofer.de, vignesh.srinivasan@hhi.fraunhofer.de, wojciech.samek@hhi.fraunhofer.de

approach remain unexplored. SSL represents an interesting avenue of research, given that unlabeled data is easily available (i.e. video sequences) and labeled data is scarce (i.e. video sequences in addition to ground-truth force data). For this purpose, Unsupervised Learning (UL) techniques are essential. They allow learning meaningful representations from unlabeled raw data. In this context, an approach based on Generative Adversarial Networks (GAN) [11] has not yet been investigated for VBFS in robotic-assisted surgery. GANs are generative models that consist in two competing neural networks with different objectives: a generator G and a discriminator D. The goal of G is to "fool" D by learning to generate samples that resemble the real data (ground-truth). In contrast, the goal of D is to distinguish between real (ground-truth) and fake (samples generated by G) data. As the training process evolves, G learns the probability distribution of the real data. Auto-encoders are neural networks with fully connected layers that encode high-dimensional data into a latent space and decode this information, reconstructing the input data in its output [12][13]. A better model for processing data with spatial correlations (i.e. images) is a Convolutional Auto-Encoder (CAE). In [14], this model is studied for feature vector extraction and pre-training of Convolutional Neural Networks (CNN). The authors concluded that this model can learn biologically plausible filters. In addition, it was found that optimizing a pre-trained CNN tends to outperform the same model with its parameters initialized from scratch. When few labeled data is available, pre-training CNNs with an UL approach can help in designing models in a SL setting. Regarding the reconstructed data, its quality is affected by the design of the loss function used to optimize the CAE model. To improve this quality and learn better representations in the latent space, the traditional CAE model can be extended to a GAN framework. For this purpose, a CAE model can be designed and optimized based on several design choices described in [15], [16], [17], [18] and [19]. An adversarial auto-encoder is proposed in [15] which shapes the distribution of the latent space using a GAN framework. The authors in [16] designed a generative model as a combination of a variational auto-encoder and a GAN framework. In particular, they pointed out that a loss function designed with an element-wise metric (i.e. squared error) is simple but not suitable for image data. Instead, they suggest a feature-wise metric to measure image similarity. Specifically, they designed a loss function that uses the hidden representations of the layers in the GAN discriminator, which improves the quality of the generated samples. The task of image-to-image translation is addressed in [17]. In that work, the authors suggest the design of a discriminator network that processes image patches and outputs a probability map instead of a single scalar value (as defined in the original GAN framework [11]). This approach improves the quality of the samples rendered by the investigated model. Furthermore, in this work, low and high frequencies of image data are modeled by a loss function designed based on the $L1$-loss and GAN framework, respectively. This loss function design avoids the blurring effect on images rendered by the model due to the $L1/L2$-loss. This approach has also been investigated in the task of video frame prediction [18]. In [19], a GAN framework is proposed for transferring the texture of real into simulated images, while preserving the annotations of simulated images. The discriminator network used in this work shares some similarities with the model described in [17], suggesting that a discriminator network that outputs a probability map represents a suitable design choice for some applications.

In the present work, a model in a SSL setting is proposed for the estimation of forces in the context of robotic surgery. This model is composed of an encoder network serially connected with an LSTM network. It addresses the estimation of forces related to pushing actions (i.e. pressing the surgical tool against soft-tissues), which are essential in the execution of tasks such as the palpation of soft-tissues. The model is optimized in two stages. First, in the UL stage, a CAE is optimized in an adversarial framework using a large dataset of unlabeled video sequences describing interactions between surgical instruments and artificial soft-tissues. The CAE design and optimization is based on the works described in [15]-[19]. The objective of this stage is to design an encoder network as a feature extractor. The feature vectors computed by this neural network represent a learned representation of high dimensional data, such as video sequences. Subsequently, this encoder network is serially connected with an LSTM network and trained in a SL setting using fewer data than in the UL stage. In this stage, video sequences in addition to ground-truth force and tool data (i.e. surgical tool trajectory and grasper status) are available. The main contributions of this work are:

- In the SL stage, the impact of applying image processing operations to video sequences, such as mean normalization and space-time transformation, in the estimated force signal quality is investigated. This study shows the importance of highlighting motion in video sequences due to tool-tissue interactions.
- The effectiveness of using a loss function with two terms is investigated in the optimization of the model in the SL stage. The first term measures the distance between the ground-truth and estimated force signals (i.e. measured by root mean squared error), while the second term measures the distance between their gradients (i.e. derivative of the force signal with respect to time). This loss function design eases the modeling of smooth and sharp details found in force and torque signals.

II. METHODS

A. Dataset

Datasets addressing the force estimation task are scarce, therefore, the experiments have been validated in a custom dataset. It consists of video sequences, tool data and ground-truth interaction forces. An experimental platform was used for this purpose. In this platform, a slave robot manipulator (Stäubli RX60B) with an attached (motorized) surgical tool interacts with a digestive apparatus made of artificial soft-tissue (Silicone-Smooth On ECOFLEX 0030). Forty-four video sequences (480×640 @ 50 FPS), totaling 4.31 hours, were recorded using 4 digital cameras (DFK 72BUC02). The tool data is described by the surgical tool-tip trajectory in the 3D space and its grasping status (i.e. opened/closed grasper), at each time instant. The interaction forces and torques between the surgical tool and artificial soft-tissues were acquired by a 6D force sensor (ATI Gamma SI-32-2.5). The force sensor resolution is: 0.00625 N for f_x and f_y, 0.0125 N for f_z, and 0.0005 Nm for all the torques (τ_x, τ_y, and τ_z). This sensor was attached at the robot manipulator's end-effector and its z-axis was aligned with the surgical instrument shaft (see Fig. 3c). The measured forces and torques lie in the range +2.5/-10 N and +/-5 Nm, respectively.

B. *Preprocessing of Video Sequences*

The recorded video sequences were processed by tracking and extracting a region of interest of size 200×300 pixels from every frame. These image regions improve the visibility of the interaction between the surgical tool and soft-tissues. With this aim, mean normalization and space-time transformations were used. The mean frame normalization consists in computing a mean frame for every video sequence by averaging all the raw frames (with equal contribution). Each computed mean frame is subtracted from every corresponding video sequence. The result of this operation is the elimination of the static background present in video sequences, since it does not contribute to the learning process in the force estimation task. In the application of human pose estimation from video sequences (with static background), this operation was found beneficial [20]. The space-time transformation consists in creating a 3-channel image by concatenating gray-scale versions of the past, current, and next color frames along the channel dimension. This image representation encodes temporal information and was found useful in the estimation of sound from silent video sequences [21].

The three types of video frames investigated in the experiments are illustrated in Fig. 1. These are raw, space-time and full-processed frames. Raw video frames processed with the space-time transformation, referred to as space-time frames, emphasize motion from two sources: (i) camera motion and (ii) motion due to the interaction between surgical instruments and soft-tissues. In this representation, image regions where motion is present are rendered as colored pixels. In contrast, static image regions are shown as pixels in gray-scale. Full-processed video frames, with mean normalization and space-time transformation, only emphasize the motion caused by the interaction of the surgical tool with soft-tissues. In this representation, only this type of motion is rendered as colored pixels, while the rest of the image is suppressed (i.e. constant gray color).

C. *Semi-Supervised Model*

The SSL model, composed of an encoder network and a LSTM network, is designed in two stages.

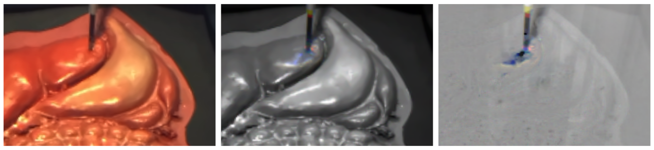

Fig. 1: Video frames investigated in the experiments. Left: Raw frames. Middle: Space-time frames. Right: Full-processed frames.

In the first stage, the encoder network is designed by optimizing a CAE in an adversarial framework as detailed in Fig. 2a. The encoder network Enc maps an input image X to a latent space vector Z. This encoding process is expressed as $Z = Enc(X)$. The reverse operation is carried out by the decoder network Dec, whose objective is to reconstruct the original image X from the latent space Z. Thus, the reconstructed image is $\tilde{X} = Dec(Z)$. The CAE is optimized in an adversarial framework using two discriminators. The first discriminator, represented by $Disc_x$, is a fully convolutional neural network that distinguishes between real (X) and reconstructed images ($\tilde{X}$). The output of this discriminator is a 3 channel probability map, $P_X \in \Re^{7 \times 10 \times 3}$. The second discriminator, $Disc_z$, is a fully connected neural network that outputs a single scalar probability, $P_Z \in \Re$. It classifies latent space vectors Z as belonging to a standard normal distribution, $Z_{real} \sim \mathcal{N}(0, 1)$, or as generated by the encoder network, $Z = Enc(X)$. In Fig. 2a, the number of output feature maps corresponding to each convolutional layer are shown for Enc (CE1-CE5), Dec (CD1-CD5) and $Disc_x$ (C1-C6). For instance, CE1-64 indicates that layer CE1 outputs 64 feature maps. Analogously, the size of each fully connected layer is shown for Enc (FCE), Dec (FCD) and $Disc_z$ (FC1-FC3). Thus, FC1-4096 describes layer FC1 with a dimension of 4096. Further details of the architecture depicted in Fig. 2a are provided in Table I. In the second stage, the model shown in Fig. 2b referred as Encoder-LSTM network, is trained end-to-end in a SL setting. This model consists in three neural networks: Enc_{video}, Enc_{tool} and Φ_{LSTM}. Enc_{video} and Enc_{tool} process video frames X_t and tool data X_t^{tool}, at each time instant t, respectively. $X_t^{tool} = [x_t, y_t, z_t, s_t]$ describes the tool trajectory in 3D space (x_t, y_t, z_t) and its grasping status s_t ($s_t = 0$ if the grasper is closed, otherwise $s_t = 1$). Enc_{video} has the same topology as Enc in the UL model depicted in Fig. 2a. Nonetheless, some changes are introduced in the layers of Enc_{video} while preserving the learned parameters from the UL stage. Enc_{tool} is a fully connected neural network that maps tool data from a lower ($X_t^{tool} \in \Re^4$) to a higher dimensional space ($Z_t^{tool} \in \Re^{64}$). It avoids the use of a very small representation for the tool data vector (i.e. $Z_t^{tool} = X_t^{tool} \in \Re^4$) with respect to that computed from video sequences ($Z_t^{video} \in \Re^{4096}$). The size of each network's layer, FC1-FC3, is indicated in Fig. 2b (i.e. FC1-16 describes layer FC1 with a dimension of 16). Table I details the design of each layer in Enc_{video} and Enc_{tool}. The neural networks Enc_{video} and Enc_{tool} output the feature

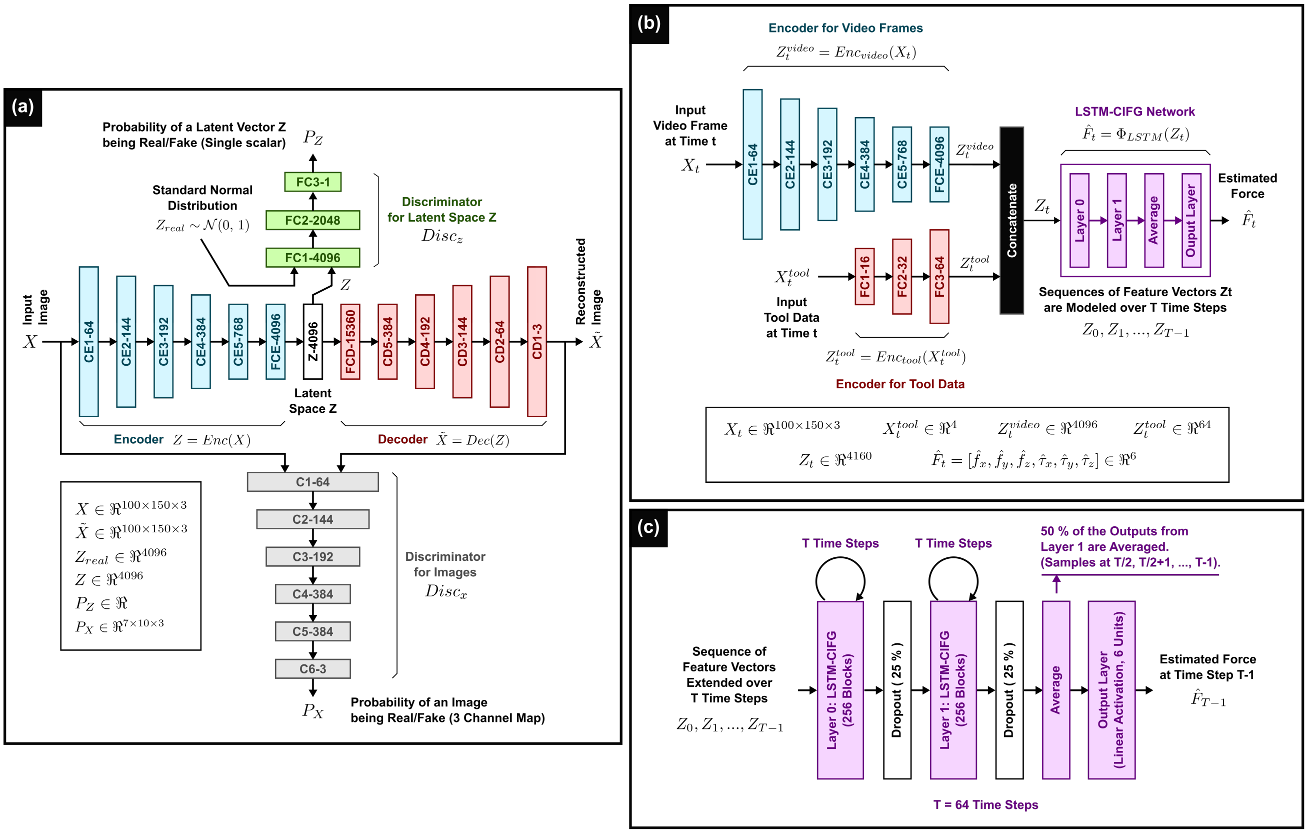
(a)
Probability of a Latent Vector Z being Real/Fake (Single scalar)
P_Z
FC3-1
FC2-2048
FC1-4096
Standard Normal Distribution
$Z_{real} \sim \mathcal{N}(0,1)$
Discriminator for Latent Space Z
$Disc_z$
Input Image
X
CE1-64
CE2-144
CE3-192
CE4-384
CE5-768
FCE-4096
Z-4096
Z
FCD-15360
CD5-384
CD4-192
CD3-144
CD2-64
CD1-3
Reconstructed Image
$\tilde{X}$
Latent Space Z
Encoder $Z = Enc(X)$
Decoder $\tilde{X} = Dec(Z)$
C1-64
C2-144
C3-192
C4-384
C5-384
C6-3
Discriminator for Images
$Disc_x$
P_X
Probability of an Image being Real/Fake (3 Channel Map)
$X \in \Re^{100\times150\times3}$
$\tilde{X} \in \Re^{100\times150\times3}$
$Z_{real} \in \Re^{4096}$
$Z \in \Re^{4096}$
$P_Z \in \Re$
$P_X \in \Re^{7\times10\times3}$
(b)
Encoder for Video Frames
$Z_t^{video} = Enc_{video}(X_t)$
Input Video Frame at Time t
X_t
CE1-64
CE2-144
CE3-192
CE4-384
CE5-768
FCE-4096
Z_t^{video}
Concatenate
Z_t
LSTM-CIFG Network
$\hat{F}_t = \Phi_{LSTM}(Z_t)$
Layer 0
Layer 1
Average
Ouput Layer
Estimated Force
$\hat{F}_t$
X_t^{tool}
Input Tool Data at Time t
FC1-16
FC2-32
FC3-64
Z_t^{tool}
$Z_t^{tool} = Enc_{tool}(X_t^{tool})$
Encoder for Tool Data
Sequences of Feature Vectors Zt are Modeled over T Time Steps
$Z_0, Z_1, ..., Z_{T-1}$
$X_t \in \Re^{100\times150\times3}$
$X_t^{tool} \in \Re^4$
$Z_t^{video} \in \Re^{4096}$
$Z_t^{tool} \in \Re^{64}$
$Z_t \in \Re^{4160}$
$\hat{F}_t = [\hat{f}_x, \hat{f}_y, \hat{f}_z, \hat{\tau}_x, \hat{\tau}_y, \hat{\tau}_z] \in \Re^6$
(c)
T Time Steps
T Time Steps
50 % of the Outputs from Layer 1 are Averaged. (Samples at T/2, T/2+1, ..., T-1).
Sequence of Feature Vectors Extended over T Time Steps
$Z_0, Z_1, ..., Z_{T-1}$
Layer 0: LSTM-CIFG (256 Blocks)
Dropout (25 %)
Layer 1: LSTM-CIFG (256 Blocks)
Dropout (25 %)
Average
Output Layer (Linear Activation, 6 Units)
Estimated Force at Time Step T-1
$\hat{F}_{T-1}$
T = 64 Time Steps

Fig. 2: (a) In the UL stage, an encoder network Enc is designed by optimizing a CAE in an adversarial framework. An input image X is transformed into a latent space Z by the encoder network Enc. The decoder network Dec reconstructs the input image X from Z, rendering $\tilde{X}$ as output. The discriminator networks $Disc_x$ and $Disc_z$ are applied on image data (X and $\tilde{X}$) and latent space (Z), respectively. (b) Encoder-LSTM model used in the SL stage. The neural networks Enc_{video} and Enc_{tool} process video frames and tool data, respectively. Their outputs are used to create a feature vector Z_t at every time instant t, which is modeled over T time steps by an LSTM-CIFG network, Φ_{LSTM}. (c) Design of the two-layer LSTM-CIFG network. Each layer has 256 blocks and processes the feature vectors Z_t over $T = 64$ time steps. 50 % of the outputs from the last cell are averaged. A fully connected layer (of dimension 6) with linear activation is used as the output layer. To prevent over-fitting, dropout is applied with probability P during training at the output of each LSTM-CIFG cell.

vectors Z_t^{video} and Z_t^{tool}, respectively. Z_t^{video} and Z_t^{tool} are concatenated into a single feature vector Z_t. Thereafter, a two layer LSTM network, Φ_{LSTM}, processes a sequence of feature vectors Z_t over T time steps to render the final estimated force $\hat{F}_t$. In this model $T = 64$ time steps. The neural network described here is the LSTM network with Coupled Input-Forget Gates (LSTM-CIFG) [22]. This model has fewer parameters than the traditional LSTM network without sacrificing performance. Fig. 2c details the design of the LSTM-CIFG model used in the experiments.

D. Model Optimization

The optimization of the SSL model starts with the CAE model and ends with the Encoder-LSTM network.

1) CAE Optimization: In this stage, the CAE parameters are updated by processing unlabeled samples from the created dataset. In the following equations, the discriminators $Disc_x$ and $Disc_z$ shown in Fig. 2a are represented by D_ϕ and Dz_α, respectively. Likewise, the CAE model, $\tilde{X} = Dec(Enc(X))$, and the encoding model for the latent space, $Z = Enc(X)$, are represented by the generator networks G_θ and Gz_β, respectively. Given M_v samples from a dataset (i.e. images), the discriminator network D_ϕ updates its parameters ϕ with the loss function defined in (1). A reconstructed image $\tilde{X}$ is computed by the generator network G_θ as an encoding-decoding process. Therefore, $\tilde{X} = G_\theta(X) = Enc(Dec(X))$. G_θ updates its parameters θ with the loss function defined in (2).

$$\mathcal{L}_D(\phi) = -\frac{1}{M_v}\sum_i^{M_v}\Big(log(D_\phi(X)) + log(1 - D_\phi(G_\theta(X)))\Big) \quad (1)$$

$$\mathcal{L}_G(\theta) = -\frac{1}{M_v}\sum_i^{M_v} log(D_\phi(G_\theta(X))) \quad (2)$$

The distribution of the latent space Z is shaped to follow standard normal distribution $\mathcal{N}(0,1)$ by using an adversarial framework. The discriminator Dz_α and generator Gz_β networks have parameters α and β, respectively. Dz_α and Gz_β are optimized with to the loss functions defined in (3) and (4), respectively. The $L1$-loss expressed in (5) was selected to

TABLE I: Design of each layer in the UL & SL models.

UNSUPERVISED LEARNING MODEL	
Layer Name	Design
CAE: Encoder Enc & Decoder Dec	
CE1-CE5	CONV5 $\downarrow 2 \rightarrow$ BN $\rightarrow$ RELU
CD1-CD5	DECONV5 $\uparrow 2 \rightarrow$ BN $\rightarrow$ RELU
FCE, FCD	DENSE
Discriminator $Disc_x$	
C1-C4	CONV5 $\downarrow 2 \rightarrow$ BN $\rightarrow$ LRELU
C5	CONV5 $\rightarrow$ BN $\rightarrow$ LRELU
C6	CONV5 $\rightarrow$ SIGMOID
Discriminator $Disc_z$	
FC1, FC2	DENSE $\rightarrow$ BN $\rightarrow$ LRELU
FC3	DENSE $\rightarrow$ SIGMOID

SUPERVISED LEARNING MODEL	
Layer Name	Design
Encoder Enc_{video}	
CE1	CONV5 $\rightarrow$ RELU $\rightarrow$ MAXPOOL
CE2-CE5	CONV5 $\rightarrow$ BN $\rightarrow$ RELU $\rightarrow$ MAXPOOL
FCE	LINEAR $\rightarrow$ BN $\rightarrow$ TANH
Encoder Enc_{tool}	
FC1, FC2	DENSE $\rightarrow$ BN $\rightarrow$ RELU
FC3	DENSE $\rightarrow$ BN $\rightarrow$ TANH

CONV5: Convolution with a kernel of size 5 × 5 (same padding). Whenever indicated, downsampling ($\downarrow$ 2) is performed with a stride of 2, otherwise stride 1. DECONV5: Transposed convolution with a kernel of size 5 × 5. Upsampling ($\uparrow$ 2) is performed with a stride of 2. DENSE: Fully connected layer (without activation). BN: Batch normalization layer. MAXPOOL: Max-Pooling layer. RELU: Rectified linear activation. LRELU: Leaky RELU activation with slope of 0.2. SIGMOID (TANH): Sigmoid (Hyperbolic tangent) activation.

penalize the difference between the ground-truth X and the reconstructed images $G_\theta(X)$. This loss function produces a lower blurring effect with respect to the $L2$-loss. Equation (6) is applied to the layers of the discriminator D_ϕ. It measures the distance between the hidden representations produced in $D_\phi^{(l)}$ at a layer l, given as input ground-truth ($D_\phi(X)^{(l)}$) and reconstructed ($D_\phi(G_\theta(X))^{(l)}$) images.

$$\mathcal{L}_{Dz}(\alpha) = -\frac{1}{M_v}\sum_{i}^{M_v}\left(log(Dz_\alpha(X)) + log(1 - Dz_\alpha(Gz_\beta(X)))\right) \tag{3}$$

$$\mathcal{L}_{Gz}(\beta) = -\frac{1}{M_v}\sum_{i}^{M_v} log(Dz_\alpha(Gz_\beta(X))) \tag{4}$$

$$\mathcal{L}_{L1}(X) = \|X - G_\theta(X)\|_1 \tag{5}$$

$$\mathcal{L}_{ACT}(X;l) = \left\|D_\phi(X)^{(l)} - D_\phi(G_\theta(X))^{(l)}\right\|_1 \tag{6}$$

The total image reconstruction loss in (7), represents a linear combination of the loss functions (2), (4), (5) and (6), weighted by the scalars λ_G, λ_{Gz}, λ_{L1} and λ_{ACT}, respectively.

$$\begin{aligned}\mathcal{L}_R(X,\theta,\beta) = {} & \lambda_G\ \mathcal{L}_G(\phi) + \lambda_{Gz}\ \mathcal{L}_{Gz}(\beta) \\ & + \lambda_{L1}\ \mathcal{L}_{L1}(X) + \lambda_{ACT}\ \mathcal{L}_{ACT}(X;l)\end{aligned} \tag{7}$$

2) Encoder-LSTM Optimization: In this stage, the parameters of Enc_{video} are initialized from the pre-trained encoder network Enc of the CAE model. Therefore, during the optimization, the parameters of Enc_{video} are fine-tuned while those of Enc_{tool} and Φ_{LSTM} are optimized from scratch. Equation (8) describes the loss function used for the joint training of the Enc_{video}, Enc_{tool} and Φ_{LSTM} networks, which define the Encoder-LSTM model (see Fig. 2b). It is a linear combination of the Root Mean Squared Error (RMSE) and Gradient Difference Loss (GDL) weighted by λ_{RMSE} and λ_{GDL}, respectively. The RMSE defined in (9), penalizes the distance between ground-truth $F_t^{(i)}$ and estimated $\widehat{F}_t^{(i)}$ force components at time t, indexed by $i = 0, ..., N-1$, where N is the total number of force components. On the other hand, the GDL defined in (10) measures the distance between the gradients of ground-truth and estimated force components, referred as $\partial F_t^{(i)}/\partial t$ and $\partial \widehat{F}_t^{(i)}/\partial t$, respectively. These gradients are approximated by convolving the i-th force component $F_t^{(i)}$ with the kernel $h = [-1 \quad 0 \quad 1]$. Therefore, $\partial F_t^{(i)}/\partial t = F_t^{(i)} * h$ and $\partial \widehat{F}_t^{(i)}/\partial t = \widehat{F}_t^{(i)} * h$. Finally, in (9) and (10), samples are summed over a temporal window from $t = 0, ..., T_S$, being T_S the total number of samples.

$$\mathcal{L}(F,\widehat{F}) = \lambda_{RMSE}\ \mathcal{L}_{RMSE}(F,\widehat{F}) + \lambda_{GDL}\ \mathcal{L}_{GDL}(F,\widehat{F}) \tag{8}$$

$$\mathcal{L}_{RMSE}(F,\widehat{F}) = \sum_{t}^{T_S}\sqrt{\frac{1}{N}\sum_{i}^{N}(F_t^{(i)} - \widehat{F}_t^{(i)})^2} \tag{9}$$

$$\mathcal{L}_{GDL}(F,\widehat{F}) = \sum_{t}^{T_S}\sum_{i}^{N}\left|\left|\frac{\partial F_t^{(i)}}{\partial t}\right| - \left|\frac{\partial \widehat{F}_t^{(i)}}{\partial t}\right|\right| \tag{10}$$

III. EXPERIMENTS & RESULTS

The dataset consists of ∼780K samples split in 77% as the training set and 23% as the test set. Each sample is represented by an RGB frame, downsized from 200 × 300 to 100 × 150 pixels. In addition, every frame is provided with a 6D vector of ground-truth interaction forces and a 4D vector of tool data. This proportion of samples (unlabeled video sequences) are used in the UL stage. Afterward, in the SL stage, a subset of samples (video sequences, force and tool data) of size ∼320K and ∼40K are taken from the training and test sets, respectively. Relatively, these samples represent a proportion of 89% as the training set and 11% as the test set. However, with respect to the total size of the dataset (∼780K samples), they represent a percentage of 41% and 5% as the training and test sets, respectively. The neural network models were implemented in Tensorflow [23] and the experiments were carried out using a single NVIDIA Titan X Graphic Processing Unit.

A. Convolutional Auto-Encoder: Image Reconstruction

The CAE was optimized over 241K iterations (∼257 hours) with the Adam [24] solver, starting with a learning rate of 1×10^{-4}. In every iteration, two gradient descent updates were applied on the parameters of the generator

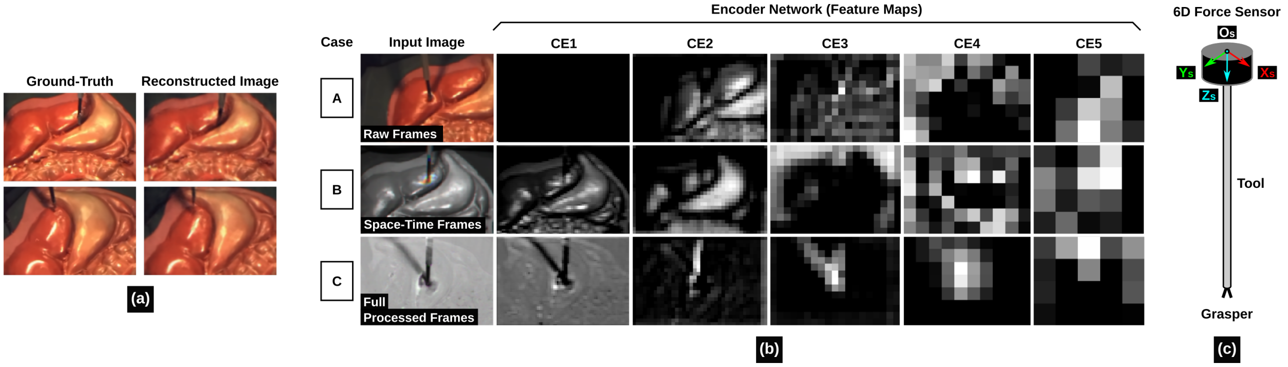

Fig. 3: (a) A sample of reconstructed images by the CAE at test time. (b) Visualization of the feature maps computed by the encoder network at each layer (CE1, ..., CE5) during the supervised learning stage. The feature maps are shown for different input video frames (corresponding to cases A, B, and C) while the network is being optimized. Setup of the 6D force sensor and surgical tool used in the experiments. During the training (inference) stage, the Encoder-LSTM network processes (estimates) force and torque data measured with respect to $O_s = \{X_s, Y_s, Z_s\}$, which is the reference frame of the force sensor with respect to the world. The force sensor z-axis, Z_s, is aligned with the tool shaft.

network G_θ, using the loss in (7). Afterward, a single update operation was performed on the parameters of the discriminator networks, D_ϕ and Dz_α, using the loss functions (1) and (3), respectively. The hyper-parameters of the loss defined in (7) are: $\lambda_{L1} = 200$ and $\lambda_G = \lambda_{Gz} = \lambda_{ACT} = 1$. Moreover, the loss function in (6) was applied to the discriminator $D_\phi^{(l)}$ at layer $l = 4$. This layer is shown in Fig. 2a as C4-384. The input video frames were corrupted with noise $\eta_f \in \Re^{100\times150\times3}$ from an uniform distribution $\mathcal{U}(0,1)$. This noise varied with intensity $\eta_i \in [0, 0.6]$ during training, according to an uniform distribution $\mathcal{U}(0,1)$. A sample of reconstructed images (test set) rendered by the CAE model are shown in Fig. 3a. In this illustration, a small blurring effect is observed in reconstructed images due to the $L1$-loss. Nonetheless, most of the image details are correctly reproduced using the adversarial framework.

B. Encoder-LSTM Network Model: Force Estimation

The Encoder-LSTM model was investigated using the three types of input data depicted in Fig. 1. This results in cases A, B and C, in which the model process raw, space-time and full-processed frames, respectively. In all cases, the Encoder-LSTM model was trained end-to-end, using Adam as optimizer and the loss function in (8) with the hyper-parameters $\lambda_{RMSE} = 1.0$ and $\lambda_{GDL} = 0.20$. The parameters of Enc_{video} were initialized from the UL stage, while the parameters of Enc_{tool} and Φ_{LSTM} were initialized from scratch (see Fig. 2b for reference). Dropout was applied to the LSTM-CIFG model with probability of $P = 0.25$ as shown in Fig. 2c. The models studied in the cases A, B and C, were optimized starting with a learning rate (exponential decay applied) of 1×10^{-3}, 5×10^{-4}, and 5×10^{-4}, completing over 86K (∼93), 109K (∼94), and 128K (∼160) iterations (hours), respectively. A fourth experiment was added to evaluate the impact of the GDL in the loss function (8) by setting, $\lambda_{RMSE} = 1.0$ and $\lambda_{GDL} = 0.0$. This results in case D, in which the model takes full-processed frames as input data (as in case C). However, the model studied in case D was optimized over 95K iterations (∼120 hours) with a learning rate of 9×10^{-4} and a dropout probability of $P = 0.30$.

TABLE II: Estimated force signal quality for each case studied (best values are highlighted in bold).

CASE†	ESTIMATED FORCE COMPONENTS					
	f_x	f_y	f_z	τ_x	τ_y	τ_z
Pearson Correlation Coefficient (PCC)						
A	0.1598	0.0370	0.1570	0.1435	0.1916	0.0899
B	0.1978	0.1457	0.1211	0.0540	0.1853	**0.1045**
C	**0.2487**	**0.2328**	**0.8084**	**0.1839**	**0.5131**	0.0585
D	0.2294	0.1097	0.7190	0.1392	0.0486	0.0723
Root Mean Squared Error (RMSE)						
Units	N			Nm		
A	0.0615	0.0593	1.2825	0.1456	0.1577	0.0160
B	**0.0553**	**0.0397**	1.3439	0.1401	0.1589	0.0120
C	0.0562	0.0406	**0.8929**	**0.1232**	**0.1332**	**0.0118**
D	0.0630	0.0436	1.0099	0.1373	0.1639	0.0133

† The models studied in cases A, B and C process raw, space-time and full-processed frames, respectively. The RMSE and GDL are considered in the loss function. In case D, the model takes as input the full-processed frames and only considers the RMSE in the loss function.

The estimated force signal quality (test set) measured by the Pearson Correlation Coefficient (PCC) and Root Mean Squared Error (RMSE), for each case (A-D), is presented in Table II. These results suggest that the Encoder-LSTM model has difficulties in learning from raw (case A) and space-time (case B) frames. This problem is alleviated by explicitly providing the Encoder-LSTM model information about the interaction between the surgical instrument and soft-tissues. Such information is emphasized in the model investigated in case C by using raw frames processed with mean normalization and space-time transformation. Nonetheless, other techniques with similar objective can potentially work. For instance, the use of optical flow or an attention model. By inspecting some of the feature map activations in the encoder network during the training stage, as depicted in Fig. 3b, it is possible to understand how difficult is to process raw and space-time frames. This illustration reveals that meaningful information, specifically motion due to tool-tissue interactions, is only propagated through the encoder network layers when this model is fed with full-processed frames (see bottom row in Fig. 3b). Regarding the loss function design, by comparing cases C and D, it is clear that using the GDL in the loss function provides advantages in the learning process. This result indicates that the RMSE and GDL ease the modeling of smooth and sharp details found in force signals, respectively.

It is important to notice that the interaction forces reported in the experiments are mainly transmitted along the surgical instrument shaft while performing pushing actions. During data acquisition, the z-axis of the force sensor was aligned with the surgical tool shaft (see Fig. 3c for reference). Therefore, in Table II, the PCC and RMSE values attributed to the estimated force f_z are the most representative. The quality of this force component, which corresponds to case C, is shown in Fig. 4b by plotting the estimated vs the ground-truth data samples. For the same case, a sample of force signals computed over time by the Encoder-LSTM model is presented in Fig. 4a. In this illustration, the amplitude of the estimated force f_z differs (at some points over time) with respect to ground-truth data. However, its shape is almost completely recovered. The rest of forces and torques have smaller values as a result of the pushing actions.

An insight from the proposed approach is that the LSTM-CIFG network is performing a time series estimation from a latent ($Z_t \in \Re^{4160}$) to a force data space ($\widehat{F}_t \in \Re^6$). From that perspective, the initial samples estimated by this network should have a large error. However, Fig. 4a shows that such samples are close to the ground-truth data (i.e. see the force component f_z). This result can be explained by the initial state of the tool in the recorded dataset. That is, the tool is not in contact with soft-tissues, and therefore, the force close to zero.

The force sensing accuracy, usually measured with the RMSE, is reported to fall below 0.1 N, both in prototyped sensors [25] and in those developed under a vision-based approach [9]. In terms of this metric, the proposed model needs to be improved for real operational purposes (see the RMSE reported for f_z corresponding to case C in Table II).

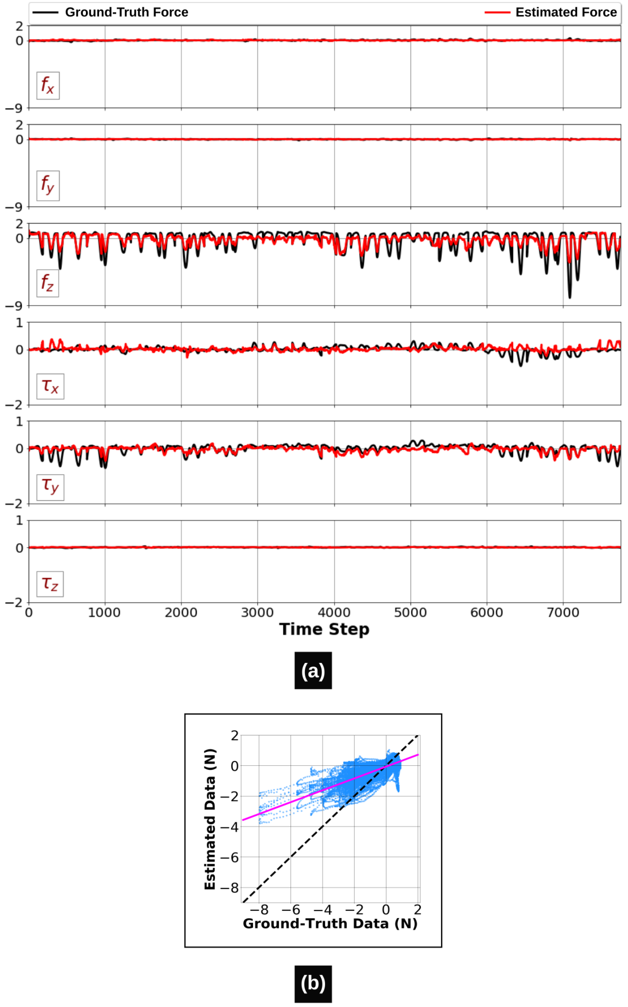

Fig. 4: (a) Estimated force signals over time for a pushing action. The amplitude of these signals is shown in force (N) and torque units (Nm). (b) Estimated vs ground-truth force data (in Newtons) related to the f_z component. The ideal line fitting the data samples (circles colored in blue) is shown in dotted style and black color. The best fitting line with a correlation coefficient of $R = 0.8084$, is depicted in solid style and magenta color.

IV. CONCLUSIONS

In this work, a VBFS model designed in a SSL setting has been investigated. The results from the UL stage, suggest (qualitatively) that the CAE model designed in an adversarial framework, provides reconstructed images with sharp details (Fig. 3a). However, some difficulties appear in the learning process during the SL stage, when the encoder network of the CAE is serially connected with the LSTM-CIFG network. The encoder network, used as a feature vector extractor, has difficulties in finding a good representation of raw and space-time frames, useful for its processing by the LSTM-CIFG network. The best results are obtained when raw frames are processed with mean normalization and space-time transformation. Therefore, this suggests the importance of providing the neural network with information about the motion that results from the interaction between the surgical tool and soft-tissues. Additionally, in the SL stage, a loss function that considers the distance between ground-truth and estimated force (i.e. using the RMSE) is not enough to provide force estimates with good quality. By taking into account the distance between the gradients of ground-truth and estimated force (i.e using the GDL), the quality of force estimates is improved. As future work, three research directions can be explored to improve the accuracy of the proposed approach. First, the use of depth information could help to reduce the gap between the amplitude of ground-truth and estimated force signals, i.e. using a method such as [26]. Second, an attention model [27] would allow to automatically process those image regions that contribute to the force. Finally, the proposed model is to be improved by interpreting its predictions with methods such as layer-wise relevance propagation [28], [29].

ACKNOWLEDGMENT

The first author of this work acknowledges the Mexican National Council for Science and Technology (CONACYT) and the Mexican Secretariat of Public Education (SEP) for their support in doctoral studies. This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 779813 (SARAS). We thank the NVIDIA Corporation for supporting this research by donating a Titan X GPU card.

REFERENCES

[1] B. Bayle, M. Joinié-Maurin, L. Barbé, J. Gangloff, and M. de Mathelin. *Robot Interaction Control in Medicine and Surgery: Original Results and Open Problems*, pages 169–191. Springer New York, New York, NY, 2014.

[2] Arturo Marbán, Alicia Casals, Josep Fernández, and Josep Amat. *Haptic Feedback in Surgical Robotics: Still a Challenge*, pages 245–253. Springer International Publishing, 2014.

[3] Christopher W Kennedy and Jaydev P Desai. A vision-based approach for estimating contact forces: Applications to robot-assisted surgery. *Applied Bionics and Biomechanics*, 2(1):53–60, 2005.

[4] E. Noohi, S. Parastegari, and M. efran. Using monocular images to estimate interaction forces during minimally invasive surgery. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4297–4302, Sept 2014.

[5] W. Kim, S. Seung, H. Choi, S. Park, S. Y. Ko, and J. O. Park. Image-based force estimation of deformable tissue using depth map for single-port surgical robot. In *Control, Automation and Systems (ICCAS), 2012 12th International Conference on*, pages 1716–1719, Oct 2012.

[6] Stamatia Giannarou, Menglong Ye, Gauthier Gras, Konrad Leibrandt, Hani J. Marcus, and Guang-Zhong Yang. Vision-based deformation recovery for intraoperative force estimation of tool–tissue interaction for neurosurgery. *International Journal of Computer Assisted Radiology and Surgery*, 11(6):929–936, 2016.

[7] A. I. Aviles, A. Marban, P. Sobrevilla, J. Fernandez, and A. Casals. A recurrent neural network approach for 3d vision-based force estimation. In *2014 4th International Conference on Image Processing Theory, Tools and Applications (IPTA)*, pages 1–6, Oct 2014.

[8] A. I. Aviles, S. M. Alsaleh, P. Sobrevilla, and A. Casals. Force-feedback sensory substitution using supervised recurrent learning for robotic-assisted surgery. In *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1–4, Aug 2015.

[9] A. I. Aviles Rivero, S. M. Alsaleh, J. K. Hahn, and A. Casals. Towards retrieving force feedback in robotic-assisted surgery: A supervised neuro-recurrent-vision approach. *IEEE Transactions on Haptics*, PP(99):1–1, 2016.

[10] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, Nov 1997.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[12] G. E. Hinton and R. R. Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

[13] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1096–1103, New York, NY, USA, 2008. ACM.

[14] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. *Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction*, pages 52–59. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[15] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian Goodfellow. Adversarial autoencoders. In *International Conference on Learning Representations*, 2016.

[16] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML'16, pages 1558–1566. JMLR.org, 2016.

[17] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *arXiv preprint arXiv:1611.07004*, 2016.

[18] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. *CoRR*, abs/1511.05440, 2015.

[19] Ashish Shrivastava, Tomas Pfister, Oncel Tuzel, Josh Susskind, Wenda Wang, and Russ Webb. Learning from simulated and unsupervised images through adversarial training. In *CVPR*, 2017.

[20] T. Pfister, K. Simonyan, J. Charles, and A. Zisserman. Deep convolutional neural networks for efficient pose estimation in gesture videos. In *Asian Conference on Computer Vision (ACCV)*, pages 538–552. Springer, 2014.

[21] A. Owens, P. Isola, J. McDermott, A. Torralba, E. H. Adelson, and W. T. Freeman. Visually indicated sounds. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2405–2413, 2016.

[22] K. Greff, R. K. Srivastava, J. Koutnk, B. R. Steunebrink, and J. Schmidhuber. Lstm: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, PP(99):1–11, 2016.

[23] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, et al. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283, 2016.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2014.

[25] U. Kim, D. H. Lee, W. J. Yoon, B. Hannaford, and H. R. Choi. Force sensor integrated surgical forceps for minimally invasive robotic surgery. *IEEE Transactions on Robotics*, 31(5):1214–1224, Oct 2015.

[26] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 270–279, 2017.

[27] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057, 2015.

[28] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7):e0130140, July 2015.

[29] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1–15, 2018.