Multi-Stream Deep Architecture for Surgical Phase Recognition on Multi-View RGBD Videos

Andru Putra Twinanda[*,1], Pramita Winata[*,1], Afshin Gangi[1,2], Michel de Mathelin[1], Nicolas Padoy[1]

[1]ICube Laboratory, University of Strasbourg, CNRS, IHU Strasbourg, France.
[2]Interventional Radiology Department, University Hospital Strasbourg, France.

Abstract. With the increasing volume of surgeries and the advancement in medicine and technology, having a context-aware system (CAS) is becoming a necessity in the operating room (OR). By understanding the surrounding physical environment, a CAS will allow the possibility to improve OR scheduling, to design context-sensitive user interfaces, and to develop the automatic transcription of medical procedures. In this paper, we address one of the essential components of a CAS, namely the ability to recognize surgical phases during a surgical procedure. Here, we focus on performing this recognition task on vertebroplasty procedures recorded by a ceiling-mounted multi-view RGBD camera system. Instead of using hand-crafted visual features, we propose to learn multi-modal visual features using deep learning techniques. We design a neural network architecture which takes RGB, depth, and motion images as input and computes a visual feature representation shared among the modalities. Using this network, visual features are then extracted and passed to a recognition pipeline, which consists of SVM and HHMM. This pipeline is used to enforce the temporal constraints from surgical workflow into the recognition process. To investigate the performance and generalizability of the network, we perform the task on two new multi-view RGBD datasets, capturing in total 37 surgeries performed in two different hybrid ORs. Through an extensive comparison with other visual features, we show that the features extracted from the proposed network yield state-of-the art results for this recognition task.

Keywords: Context-aware system, phase recognition, multi-view RGBD system, deep architecture, and neural network.

1 Introduction

In recent years, with the increasing volume of surgeries and the advancement in technology, the operating room (OR) has become a dense working environment. Equipped with advanced surgical equipments, the OR is overflown with information coming from these devices, which can impede the effectiveness and efficiency

[*] The first two authors contributed equally to this work.

in the execution of surgical procedures. Therefore, there is a growing interest in the community to construct a context-aware system (CAS) for the OR in order to be able to exploit the information to the clinicians' and surgeons' advantages [1]. A key component of a CAS is the capability to know at any time what is happening in the OR, especially during a surgery. This task, referred to as *surgical phase recognition*, consists of determining the surgical phase occurring in the OR at any given time during a surgery. The ability to recognize the surgical phases in the OR opens up the possibilities for various applications. For example, by knowing what phase is occurring in the OR, the appropriate information can be displayed to the surgeons and the clinical staff, which improves the efficiency of coordination and communication in the OR. Furthermore, through further analysis of the phases, surgical error and probable upcoming complications could potentially be avoided.

In this paper, we study the task of surgical phase recognition on RGBD videos which capture the scene in the OR. Specifically, we address the task of recognizing 8 surgical phases (see Section 3.1) in vertebroplasty procedures. The procedure consists of injecting a special cement into a fractured vertebra, with the goal of relieving spinal pain and restoring mobility. The RGBD videos are recorded using a ceiling-mounted multi-view RGBD system, which consists of two cameras. A multi-view system is used in order to cover a larger area inside the OR, while the RGBD sensors are chosen because they provide complementary color and depth information about the scene.

In the literature, multiple studies have proposed vision-based methods to perform the activity recognition task in medical settings. For example, one of the earliest work [2] proposed to identify four OR occupancy states using the videos recorded by a ceiling mounted camera. Another work [3] presented a pipeline to perform the automatic transcription of trauma resuscitations in the emergency department, using a camera mounted on top of the patient's bed. In [4], a method for activity recognition in an intensive care unit (ICU) using an RGBD sensor was presented. However, all the afore-mentioned studies address the task using *handcrafted* visual features. These features are engineered to capture certain characteristics from the data, which may lead to information loss. In this work, we are interested in performing the task using feature learning techniques in the context of multi-view multi-modal data, particularly using deep learning algorithms.

In the computer vision community, deep learning methods, such as convolutional neural networks (CNNs), have been shown to successfully perform various tasks, such as image classification [5], object detection [6], and activity recognition [7]. The methods have also been shown to successfully perform several tasks on multi-modal data. For example, in [8], the combination of RGB and depth features extracted using deep networks are shown to perform better than the individual feature for an object recognition task. Another work [9] proposed to combine RGB and motion information as input for the CNN and demonstrated better performance for activity recognition compared to other networks with single modality. Inspired by these methods, here, we investigate the usage of deep

learning techniques for feature learning on multi-modal multi-view data, by designing a multi-stream CNN architecture which takes color, depth and motion images as input.

Training a CNN is however not trivial since it typically contains millions of unknowns. For example, the AlexNet network [5] contains over 60M unknowns. This leads to the need for a large amount of data during the training process. This problem can however be alleviated thanks to *transfer learning* approaches, such as *fine-tuning*. To train the network, the fine-tuning process initializes the optimization with a successfully pre-trained network, instead of a random initialization. This significantly helps the optimization process and leads to faster convergence, as demonstrated for instance in [6] for an object detection task. In a recent work [10], it has also been shown that fine-tuning can be used to successfully train a CNN model for surgical phase recognition on laparoscopic videos.

Here, we are performing a study similar to [10], using a different type of data. We propose a training strategy for the multi-stream network in order to improve the performance of the network. Once the network is learnt, it will be used to extract the visual features from the images. These features are passed to a recognition pipeline, which consists of a support vector machine (SVM) and a hierarchical hidden Markov model (HHMM). The objective of this pipeline is to enforce the temporal constraints of the surgical workflow.

To validate our approach, we train and evaluate the network using a multi-view multi-modal dataset recorded in a hybrid OR, containing 24 surgeries. Ultimately, to show the generalizability of the network, we also perform the recognition task on another dataset of 13 surgeries, which is recorded in a different OR, without retraining the network.

The main contributions of our work are as follows: (1) we propose a CNN architecture to address phase recognition on multi-view RGBD videos and present a strategy to fine-tune the network on multi-modal data, and (2) we perform an extensive experiment to show a wide range of comparisons between the proposed network and other methods.

2 Methodology

2.1 Architecture

The proposed architecture is inspired by AlexNet [5] (shown in Fig. 1), which consists of an input layer, five convolution layers, and three fully-connected layers. The input layer takes RGB images and the output is 1000 values representing the confidence of the images belonging to the corresponding 1000 classes.

In this work, we are not only working with RGB images, but also depth images. Therefore, here we design a network which is optimized to perform surgical phase recognition using both RGB and depth images at the same time. In [9], it has been shown that the activity recognition results are improved when motion images are incorporated into the neural network as input. In addition, a study

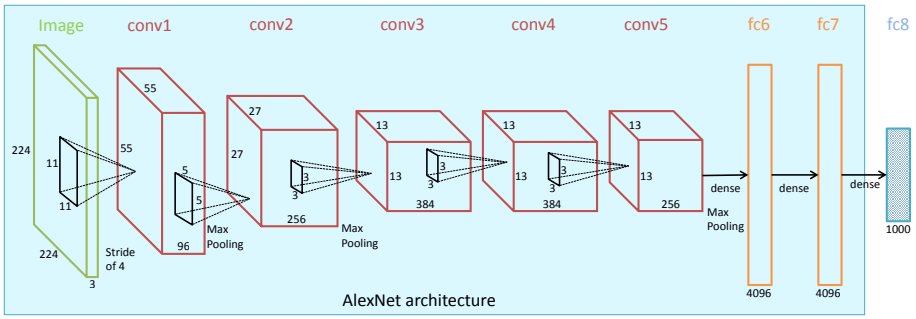


Fig. 1. AlexNet architecture [5].

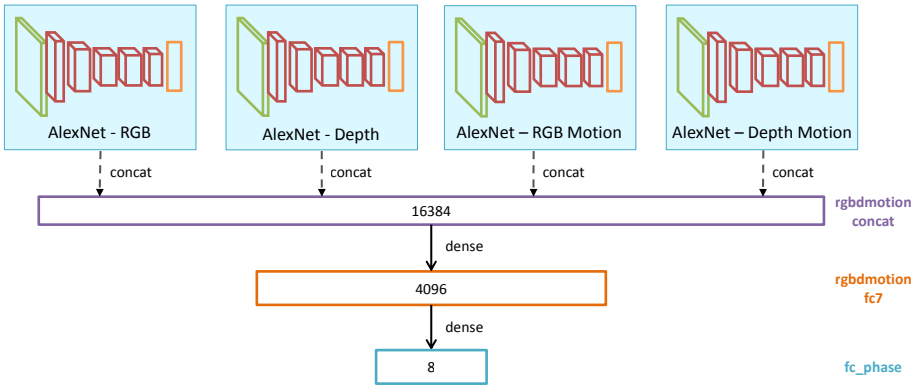


Fig. 2. Our proposed multi-stream CNN architecture.

[11] addressing a similar task to the one presented in this paper has shown that the location of the movements in the scene is one of the discriminative features. Therefore, we also incorporate motion images as input for the network. Here, motion images are obtained from both RGB and depth videos by subtracting the image at time t with image at time $t-1$.

Taking the afore-mentioned points into consideration, we design a CNN architecture which is shown in Fig. 2. The network takes four streams of input, i.e., RGB, depth, and their corresponding motion images. These networks are connected to a concatenation layer after the first fully-connected layer (i.e., `fc6`). We do not perform the concatenation after layer `fc7` because we want to build a shared feature representation between the image modalities before going to the last layer (i.e., `fc_phase`) without adding another fully-connected layer. Therefore, in our proposed network, each image still undergoes the same process as in AlexNet: five convolution and three fully-connected layers. To handle the multi-view images, this network is trained using images from both views. Once trained, the network is then used to extract the visual features from images from both views.

2.2 Training Strategy

As shown in Fig. 2, it can be seen that the multi-stream network is extremely large, containing 20 convolution layers and six fully-connected layers. Fine-tuning the network in one run might not lead to result improvement since it is difficult to perform the optimization for a large number of variables. In addition, the pre-trained AlexNet model is optimized using RGB images, thus the network weights are not optimized for other image modalities. To alleviate this problem, we propose a two-step optimization.

First, using a pre-trained AlexNet network, we finetune each network stream separately. We finetune a network solely using the RGB images and repeat the same process for depth, RGB motion, and depth motion images. **Second**, using the four networks obtained from the first step, we finetune the proposed network. This way, the optimization is initialized with a semi-optimal solution, consisting of four networks independently optimized for the surgical phase recognition task. Therefore, the main objective of this second step is to obtain the optimal weights for layers `rgbdmotionfc7` and `fc_phase`.

2.3 Phase Recognition Pipeline

We adopt the phase recognition pipeline presented in [10] which consists of a multi-class support vector machine (SVM) and a hierarchical hidden markov model (HHMM). The SVM is designed to take the feature representations of the video frames to compute the confidence values indicating the frames belonging to the phases. Here, the feature representation is taken from the second last layer of the network, i.e., the output of layer `rgbdmotionfc7`. Since we are working with a multi-view system, the feature representation of a frame is obtained by concatenating the features extracted from both views.

One can observe that the confidence values computed by SVM are similar to the ones given by the network, i.e., the output of layer `fc_phase`. Thus, in practice, it is not essential to pass the visual features to the SVM to perform the recognition task. However, the SVM step is necessary in order to facilitate fair comparisons with other visual features. In [10], it is mentioned that there is only a slight difference of performance between the confidence values computed by the SVM and the ones given by the network.

Even though the confidence values obtained from the SVM can already be used to determine the surgical phases, they are obtained frame-wise without taking into account the temporal constraints imposed by the surgical workflow. To enforce these constraints, here, we use use a two-level Hierarchical HMM. The top-level contains states that model the surgical phases and their transitions, while the bottom-level nodes model the intra-phase dependencies. The temporal model is learnt from the ground truth annotations as presented in [12]. The observations are given by the confidences from the SVM.

(a) (b)

Fig. 3. Frame samples from: (a) VerCArm24 and (b) VerCT13 datasets.

3 Experimental Setup

3.1 Datasets

To perform the task, we prepare two datasets, referred to as VerCArm24 and VerCT13, recorded in two different hybrid ORs. In Fig. 3, we show the views captured by the multi-view system in both ORs. The cameras are configured to observe the OR bed and the equipment table in both ORs. This way, all major activities occurring during surgery are captured by the recording system. The VerCArm24 dataset contains 24 recordings of vertebroplasty procedures, while the VerCT13 dataset contains 13 recordings. All recordings are annotated with 8 surgical phases defined by a senior clinician. The list and statistics of the phases are shown in Fig. 4-a and the phase transitions are illustrated in Fig. 4-b.

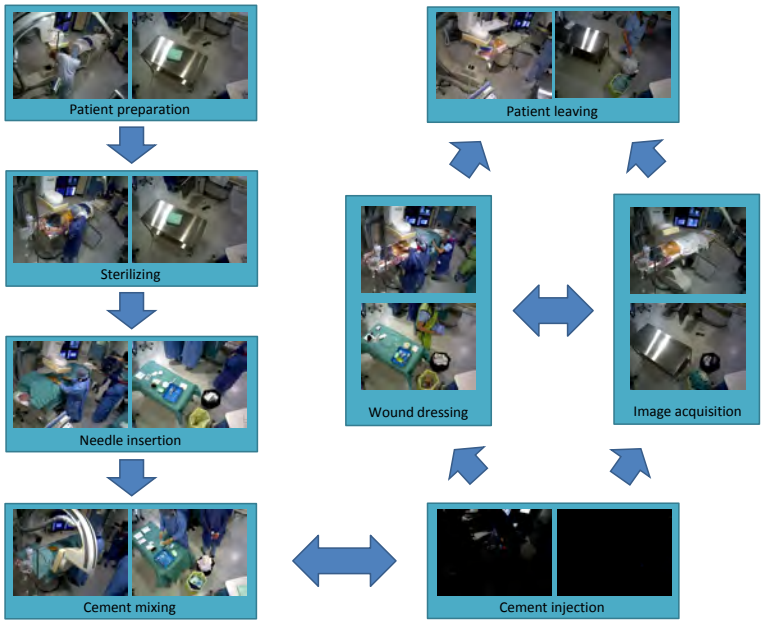
From the VerCArm24 dataset, we take 10 videos to perform the finetuning process. The rest (i.e., 14 videos) is used for evaluation. The VerCT13 dataset is used to test the generalizability of the network, thus all videos in the VerCT13 dataset are used for evaluation. In summary we have three subsets: (1) VerCArm24 finetuning, (2) VerCArm24 evaluation, and (3) VerCT13 evaluation. On both evaluation subsets, the method is evaluated using leave-one-out cross validation. For example, using the VerCArm24 evaluation subset, 13 videos are used to train the SVM-HHMM pipeline and 1 video is used for testing. Results are averaged over all possible video combinations. Due to redundancy and in order to reduce the computational cost, we perform all processes at 1 fps.

3.2 Evaluation

To perform the evaluation of the method, we use three evaluation metrics which have already been used for the same task [10], i.e., precision, recall, and accuracy. Precision and recall show the quality of the recognition results for each phase. In contrast, accuracy represents the percentage of correct detections in the complete surgery. We use three evaluation metrics in order to obtain more comprehensive results since short phases will not be well represented in accuracy since it is computed over the complete video. Due to the stochastic properties of the SVM-HHMM pipeline, we perform the evaluation in 8 experimental runs. The displayed results are obtained by averaging the results over the runs.

Phase	Length (min.)	
	VerCArm24	VerCT13
Patient preparation	16.3 ± 9.9	37.0 ± 21.7
Sterilizing	11.4 ± 3.2	12.6± 2.4
Needle insertion	18.6 ± 13.4	28.3 ± 21.9
Cement mixing	4.2 ± 1.1	3.5 ± 1.4
Cement injection	8.5 ± 2.8	18.6 ± 14.2
Wound dressing	3.6 ± 1.3	4.8 ± 1.8
Image acquisition	2.1 ± 1.0	2.8 ± 1.6
Patient leaving	3.0 ± 0.9	5.7 ± 4.5
Complete surgery	68.8 ± 30.9	116.6± 48.4

(a)



(b)

Fig. 4. (a) Phase list and statistics in VerCArm24 and VerCT13 datasets. (b) Phase transitions generated from the datasets.

We compare the performance of our proposed features with handcrafted visual features, i.e., dense SIFT on both RGB and depth images. In addition, we also perform comparisons with other deep features, such as features extracted using AlexNet [5] and finetuned networks based on AlexNet. Similarly to our proposed feature, the deep features are the output of the second last layer in the network, e.g., for AlexNet, it is the layer `fc7` (see Fig. 1). For these deep features, we also use the combination of RGB and depth features. These are obtained by concatenating the visual features extracted from the RGB and depth images.

3.3 Training Parameters

We use the same training parameters for all finetuning processes. Each of them is performed for 30K iterations with a batch size of 50 images. For layers that are already in the pre-trained network, their learning rate is set to 10^{-3}; while other layers which are initialized randomly have higher learning rate of 10^{-2}. The finetuning process is performed using the Caffe framework [13].

To carry out the phase recognition task, all features are passed to non-linear kernel SVMs, i.e., the histogram intersection kernel. For the HHMM, we set the number of top-level states to eight (equal to the number of phases), while the number of bottom level states is data-driven. To model the output of the SVM, we use a mixture of five Gaussians with diagonal covariance.

Feature	Precision (%)			Recall (%)			Accuracy (%)		
	Before	Offline	Online	Before	Offline	Online	Before	Offline	Online
DenseSIFT-RGB	72.5±5.4	87.3±14.0	82.4±13.8	77.8±5.1	87.2±13.7	84.0±13.2	74.6±6.9	86.3±16.4	81.5±15.7
DenseSIFT-Depth	62.4±6.7	72.7±13.1	69.5±11.5	63.5±9.9	72.5±13.5	69.1±10.7	66.8±19.7	80.0±11.1	73.4±17.5
AlexNet-RGB	74.4±0.8	81.5±4.2	77.9±4.5	74.5±1.1	79.6±3.9	77.5±3.8	77.3±0.9	82.9±3.1	79.6±3.2
AlexNet-Depth	60.2±0.9	77.6±3.0	73.5±1.8	56.8±1.3	77.3±2.5	74.6±1.3	59.6±1.2	78.9±2.8	75.4±2.0
AlexNet-RGBD	74.2±0.6	85.8±1.7	82.4±1.0	73.9±0.7	84.4±1.5	82.3±1.6	78.0±1.1	86.9±1.8	83.7±1.3
FTAlexNet-RGB	83.0±8.5	87.1±9.4	84.0±1.2	84.5±12.2	86.9±2.0	84.3±2.3	87.6±6.6	90.5±1.5	88.0±1.0
FTAlexNet-Depth	69.4±7.4	85.8±10.1	82.9±8.8	69.9±8.8	83.9±10.9	82.3±9.4	74.1±9.7	86.4±10.1	83.9±8.9
FTAlexNet-RGBD	82.0±0.4	88.6±0.6	84.8±0.7	82.8±0.5	87.7±0.4	84.6±0.6	86.2±0.3	91.2±0.4	88.1±0.5
Proposed Network	**86.6±5.8**	**93.3±6.9**	**91.2±5.7**	**88.9±4.2**	**91.7±5.8**	**89.6±5.7**	**91.3±4.3**	**96.0±2.8**	**93.7±3.0**

Table 1. Phase recognition results (mean±std) on the VerCArm24 dataset, including results before and after applying the HHMM (offline and online recognitions). The best result for each evaluation metric is shown in bold.

4 Experimental Results

4.1 VerCArm24 Dataset

In Table 1, we show the results of performing surgical phase recognition on the VerCArm24 dataset. Observing the results before applying the HHMM, it can be seen that deep features extracted from AlexNet perform similarly compared to the dense SIFT features. This is however expected since AlexNet is trained to extract features from completely different images, more specifically natural images. Once the network is finetuned, the performance of the features (denoted by FTAlexNet) is significantly improved. Interestingly, combining the RGB and depth deep features does not always lead to improvement. This might be due to the fact that the combination is performed through concatenation, which might lead to dimensionality problem for SVM classification. This is why in our proposed network, we propose to compute a shared visual feature representation instead of performing feature concatenation. As shown in Table 1, the best recognition results before applying the HHMM are obtained using our proposed network, yielding an accuracy of 91.3%.

Despite the high performance of the method before applying the HHMM, there is no temporal constraint incorporated into the recognition process. Once the temporal constraints are enforced by the HHMM, the recognition results are further improved. In Table 1, we show these results in offline and online columns. It can be seen that the offline recognition results are better than the online ones. This is expected due to the nature of offline recognition, where the method can see the complete sequence, while in online recognition, the method predicts the phase at time $t = t_i$ using images for time $t < t_i$. Generally, a similar trend is observed across offline and online results: our proposed network yields the best performance for both offline and online recognitions.

Feature	Precision (%)			Recall (%)			Accuracy (%)		
	Before	Offline	Online	Before	Offline	Online	Before	Offline	Online
AlexNet-RGB	69.2±0.9	76.9±3.0	71.6±2.6	69.2±0.9	72.5±2.9	68.9±2.4	78.2±0.7	82.9±1.4	78.9±1.4
AlexNet-Depth	54.6±0.4	65.3±2.6	64.0±2.7	54.5±1.2	64.5±1.6	63.6±1.7	58.6±0.6	75.2±2.1	73.3±1.6
AlexNet-RGBD	71.4±1.1	83.2±2.8	78.4±2.7	71.4±1.1	79.4±2.7	76.1±2.4	78.8±1.0	89.1±2.6	86.4±2.1
FTAlexNet-RGB	72.7±1.0	64.1±6.9	61.1±5.7	74.8±1.4	63.9±5.9	61.5±5.3	80.7±1.0	73.8±5.1	71.9±4.4
FTAlexNet-Depth	55.6±0.5	68.8±2.2	64.6±2.2	56.4±1.0	65.2±2.1	63.3±1.6	62.6±1.0	76.9±2.5	75.1±1.9
FTAlexNet-RGBD	69.1±0.2	71.4±5.1	66.0±4.5	68.1±0.1	67.4±4.5	64.8±4.2	78.1±0.3	80.8±4.5	78.0±3.9
Proposed Network	**82.2±4.9**	**89.3±6.6**	**84.8±7.4**	**83.2±6.8**	**84.8±7.4**	**82.0±6.6**	**89.0±4.7**	**95.2±2.4**	**93.9±2.2**

Table 2. Phase recognition results (mean±std) on the VerCT13 dataset, including results before and after applying the HHMM (offline and online recognitions). The best result for each evaluation metric is shown in bold.

4.2 VerCT13 Dataset

In order to test the generalizability of the networks, we also perform evaluations using another dataset, i.e., the VerCT13 dataset, that has not been seen by the networks. Here, we solely focus on the performance of deep features since it has been shown in Section 4.1 that the handcrafted features are outperformed by the finetuned deep features.

We show the phase recognition results in Table 2. It can be seen that the results hold the same trend as the ones obtained from the VerCArm24 dataset (shown in Table 1). Before applying the HHMM, our proposed network yields the best performance, with an accuracy of 89.0%. It can also be seen that the recognition results after applying the HHMM obtained by our proposed network are the best, yielding accuracies of 95.2% and 93.9% for offline and online recognitions, respectively. These results are very similar to the ones presented in Table 1, where our network yields accuracies of 96% and 93.7% for offline and online recognitions, respectively, on the VerCArm24 dataset. The fact that the features extracted from our proposed network perform similarly on the VerCT13 and the VerCArm24 datasets demonstrates that the network does not overfit the finetuning dataset, i.e., it generalizes to other datasets.

5 Conclusions

In this paper, we have presented a surgical phase recognition task performed on multi-view RGBD videos which capture the OR during vertebroplasty procedures. We propose a convolutional neural network (CNN) architecture leveraging the multi-modality of the data, by taking RGB, depth and motion images as input. To evaluate the performance, we have performed an extensive experiment using two datasets. The results demonstrate the visual features extracted by our proposed network significantly outperforms other visual features, yielding accuracies of 96% and 95.2% for offline recognition on the two datasets.

In the future work, it would be interesting to incorporate a temporal model, such as recurrent neural network (RNN) or long short term memory (LSTM)

units, into the network in order to establish an end-to-end deep architecture. This is currently still a challenging problem since it is hard to train such a model over very long sequences. Moreover, more videos might be required to properly learn the temporal model. However, incorporating the temporal model into the architecture would eliminate the need for HHMM, resulting in the possibility to optimize the full pipeline in one run.

References

1. Cleary, K., Chung, H.Y., Mun, S.K.: Or2020 workshop overview: operating room of the future. In: CARS. Volume 1268 of International Congress Series. (2004) 847–852
2. Bhatia, B., Oates, T., Xiao, Y., Hu, P.F.M.: Real-time identification of operating room state from video. In: AAAI. (2007) 1761–1766
3. Chakraborty, I., Elgammal, A., Burd, R.S.: Video based activity recognition in trauma resuscitation. 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG) **0** (2013) 1–8
4. Lea, C., Facker, J.C., Hager, G.D., Taylor, R.H., Saria, S.: 3d sensing algorithms towards building an intelligent intensive care unit. In: AMIA Summits on Translational Science. (2013)
5. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems 25. (2012) 1097–1105
6. Girshick, R.B., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR. (2014) 580–587
7. Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M.: Learning spatiotemporal features with 3d convolutional networks. In: ICCV. (Dec 2015) 4489–4497
8. Wang, A., Cai, J., Lu, J., Cham, T.J.: Mmss: Multi-modal sharable and specific feature learning for rgb-d object recognition. In: 2015 IEEE International Conference on Computer Vision (ICCV). (Dec 2015) 1125–1133
9. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. CoRR (2014)
10. Twinanda, A.P., Shehata, S., Mutter, D., Marescaux, J., de Mathelin, M., Padoy, N.: Endonet: A deep architecture for recognition tasks on laparoscopic videos. IEEE Transactions on Medical Imaging (2016)
11. Twinanda, A.P., Alkan, E.O., Gangi, A., de Mathelin, M., Padoy, N.: Data-driven spatio-temporal RGBD feature encoding for action recognition in operating rooms. Int. J. Computer Assisted Radiology and Surgery **10**(6) (2015) 737–747
12. Padoy, N., Mateus, D., Weinland, D., Berger, M.O., Navab, N.: Workflow monitoring based on 3D motion features. In: ICCV Workshops. (2009) 585–592
13. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)