applied sciences

MDPI

Article

Robust Real-Time Detection of Laparoscopic Instruments in Robot Surgery Using Convolutional Neural Networks with Motion Vector Prediction

Kyungmin Jo [1,†], Yuna Choi [2,†], Jaesoon Choi [1,*] and Jong Woo Chung [3,*]

1 Department of Biomedical Engineering, Asan Medical Center, University of Ulsan College of Medicine, Seoul 138-736, Korea
2 Department of Medicine, University of Ulsan College of Medicine, Seoul 138-736, Korea
3 Department of Otolaryngology, Asan Medical Center, University of Ulsan College of Medicine, Seoul 138-736, Korea
* Correspondence: fides@amc.seoul.kr (J.C.); jwchung@amc.seoul.kr (J.W.C.)
† These authors contributed equally to this work.

Received: 2 May 2019; Accepted: 16 July 2019; Published: 18 July 2019

Abstract: More than half of post-operative complications can be prevented, and operation performances can be improved based on the feedback gathered from operations or notifications of the risks during operations in real time. However, existing surgical analysis methods are limited, because they involve time-consuming processes and subjective opinions. Therefore, the detection of surgical instruments is necessary for (a) conducting objective analyses, or (b) providing risk notifications associated with a surgical procedure in real time. We propose a new real-time detection algorithm for detection of surgical instruments using convolutional neural networks (CNNs). This algorithm is based on an object detection system YOLO9000 and ensures continuity of detection of the surgical tools in successive imaging frames based on motion vector prediction. This method exhibits a constant performance irrespective of a surgical instrument class, while the mean average precision (mAP) of all the tools is 84.7, with a speed of 38 frames per second (FPS).

Keywords: robot surgery; tool detection; YOLO; CNN; real-time; convolutional neural networks

1. Introduction

According to the World Health Organization (WHO), complications in inpatient surgical operations occur for at most 25% of the patients, and at least half of the cases in which surgery led to harm or damage are considered preventable [1]. This means that improvement of surgical performance can lead to better outcomes of surgical operations. Surgical performance can be improved through sophisticated remote manipulation of the robot [2–4], but surgical feedback also has a positive effect on surgical performance [5]. While manual evaluation methods such as the objective structured assessment of technical skills (OSATS), and the global operative assessment of laparoscopic skills (GOALS) can assess the surgical skills and are beneficial in terms of their improvements, it is both time and labor consuming, because surgeries could last multiple hours [6,7]. Manual assessment is subjective to observer bias and can lead to subjective outcomes [8]. Detection of surgical instruments is one of the indicators used for the analysis of surgical operations and it can be useful for the effective and objective analysis of surgery [9]. This also helps prevent surgical tool collision by informing the operator during the procedure [10].

Various approaches have been published on surgical tool detection. Cai et al. [11] imaged markers, which were placed on surgical instruments with the use of two infrared cameras. Kranzfelder et al. [12] presented an approach based on radiofrequency identification (RFID) for real-time tracking of

laparoscopic instruments. However, at present, there is no proper and reliable antenna system for routine intraoperative applications [12]. However, detection tools utilizing markers interfere with the surgical workflow and require modifications of the tracked instrument [13].

Efforts have been expended to develop vision-based and marker-less surgical tool detection using feature representations, based on color [14,15], gradients [16], or texture [17]. Speidel et al. [18] segmented the instruments in images and recognized their types based on three-dimensional models. Many researchers have also addressed surgical tool detection with the use of convolutional neural networks. Putra et al. [19] proposed for the first time the use of a CNN for multiple recognition tasks on laparoscopic videos. Several works [20–22] of surgical tool detection by CNNs have been proposed as a part of the M2CAI 2016 tool presence detection challenge [23]. Jin et al. [24] performed surgical tool localization and phase recognition in cholecystectomy videos based on faster R–CNN. Bodenstedt [25] proposed a new method to detect and to identify surgical tools by calculating a bounding box using a random forest algorithm, and then, extracting multiple features from each bounding box. Shvets et al. [26] introduced a method of robotic instrument semantic segmentation based on deep learning, in both binary and multiclass settings. However, these studies dealt with tool detection in a frame-wise manner, but did not employ time information, and did not detect tools in real time.

In this study, we address the issue of tool detection in laparoscopic surgical videos. Our method is faster and more accurate than cutting-edge technologies [20–22,24], and it can be applied during surgery or real-time analyses. We propose a new method to detect the surgical tool in laparoscopic images using YOLO9000 [27] and detect missing surgical tools based on motion vector prediction.

2. Surgical Tool Detection

The proposed algorithm consists of two stages. The first step aims to detect the surgical tool used in the current frame based on YOLO9000. The second step is to check for the presence of the surgical tools that were not detected in the first step, and to detect them additionally (Figure 1).

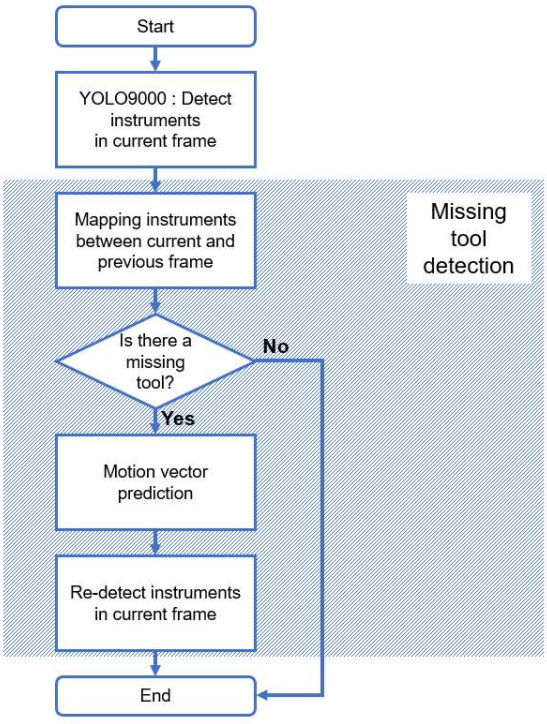


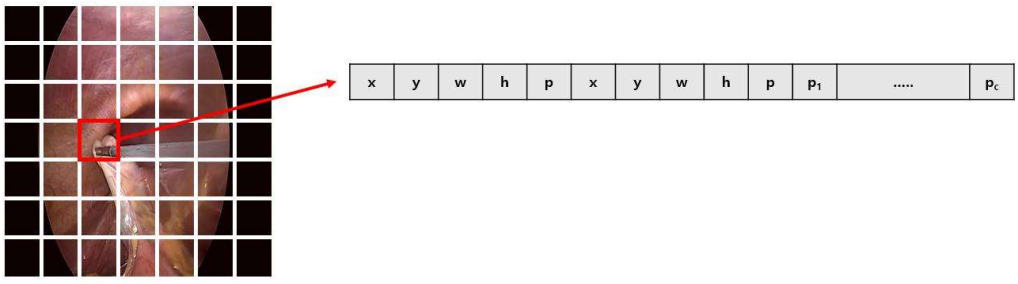
Figure 1. Flow chart of the proposed algorithm. The top rectangle is the detection step of the surgical tool in the current frame using YOLO9000, and the rest in the upper-right direction pattern corresponds to the missing tool detection step.

2.1. Detection with YOLO9000

The proposed method detects a surgical tool using YOLO9000, which is based on a convolutional neural network (CNN). CNN usually draws feature maps from input images using convolutional and pooling layers. In the convolutional layer, the filter extracts the pattern corresponding to each filter in the entire image area based on a convolution operation. Alternatively, the pooling layer generally reduces the size of the output of the preceding convolutional layer, thereby reducing the size of feature map inputs to the next layer, and consequently reduces the total number of parameters required for training. Maximum pooling and average pooling are mainly used for CNN.

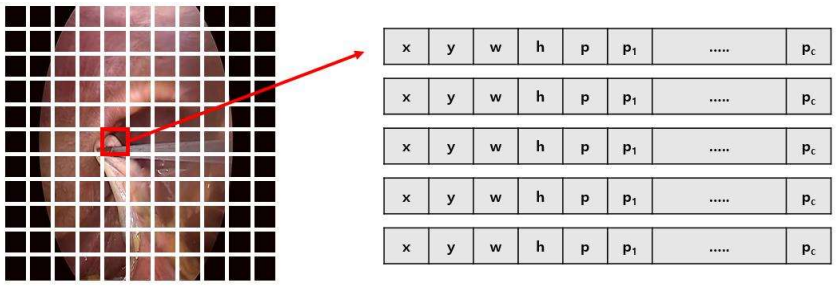
You Only Look Once (YOLO) is one of CNN-based detection methods and is suitable for real-time processing. As the general region-based CNN identifies the region-of-interest directly from every input image, it requires considerable computation to detect the position of an object. As a result, it is difficult to detect an object in real time. However, YOLO divides all input images into S × S grid cells. Additionally, the size of the bounding box is set in advance. To be specific, the bounding box is preset by clustering using the box size of the ground truth in the training dataset. Therefore, during training or testing, only B pre-defined bounding boxes are calculated for each grid cell. This is a major difference between YOLO and region-based CNNs, such as the faster R–CNN, and this is the reason why YOLO can detect objects in real time.

The algorithm we used for the purposes of this study is YOLO9000, which is the second of the three versions of YOLO. YOLO9000 uses small grid cells and changes layers to improve accuracy over the previous versions of YOLO. Figure 2 shows the difference between the first version (V1) [28] and the second version (V2) of YOLO. In V1, S is 7 and B is 2, but S is 13 and B is 5 in V2. V2 uses more bounding boxes compared to V1. This is because the size of one grid cell is reduced by increasing the number of grid cells. As a result, it is easy to detect smaller objects. In V1, the configuration of each grid cell in the last layer is (5 × B + C), but it is (5 + C) × B in V2. C is the number of classes. In V1, the probability that a grid cell corresponds to each class is calculated separately from the probability that each bounding box contains an object. By multiplying these two values, the class to which each bounding box corresponds can be determined. In V2, however, class and object probabilities are obtained for each bounding box unit. Furthermore, the fully connected layer of V1 is replaced with the convolutional layer in V2. Therefore, it is designed so that it does not lose spatial information. Finally, unlike V1, V2 uses batch normalization on the convolutional layer to enhance the learning effect in the mini batch. Leaky ReLU [29] is also applied as an activation function for nonlinearity between layers, and maximum pooling is applied. Based on the differences, we applied YOLO9000, that can better preserve the spatial location information of tools for surgical tool detection.



(a)

Figure 2. *Cont.*



(**b**)

Figure 2. Difference between the first version (**a**) and the second version (**b**) of You Only Look Once (YOLO): input image is divided into grid cells (S × S). Then, prediction of each grid cell is encoded as a tensor. x and y are the center position, and w and h are the width and height of each bounding box. p is the probability of whether it is an object or not, and p_c is the probability corresponding to each class.

Although YOLO9000 adjusts the size of the grid cell and uses identity mapping to detect small objects, it is still difficult to detect small-sized surgical tools, because the input image is resized to 416 × 416, which is typically a smaller size compared to the original image. To solve this problem, the third version of YOLO [30] detects objects at three scale levels according to the residual skip connection and upsampling. In addition, multiple label classifications are possible. As a result, the object detection ratio increases, but the computational time also increases, and the speed decreases. For this reason, the third version of YOLO is not suitable for surgical tool detection in real time. The surgical tool detection problem consists of seven classes and requires a single-label classification in real time. Therefore, in the proposed method, it is performed by applying YOLO9000, and the missing tools are additionally detected through motion vector prediction with tool mapping.

2.2. Missing Tool Detection with Motion Vector Prediction

The missing tool detection process is subdivided into the following steps—a tool mapping and a tool redetection. In the tool mapping step, the presence of a missing tool is checked. To be specific, the tools identified in the current frame (t) are compared to that of the previous frame (t−1), based on the number and class. If one or more of the tools of this frame (t) have the same class as the tools of the previous frame (t−1), the tool that is closest to the tool of the previous frame (t−1) is considered as the same tool in the current frame (t). Conversely, if a tool only exists in the previous frame (t−1), it is determined that a missing tool exists.

Once the existence of the missing tool is confirmed, motion vector prediction is performed as shown in Figure 2. As the YOLO9000 classifies the surgical tool using a predetermined bounding box, if the main feature of the surgical tool is located at the boundary of the bounding box due to the movement of the surgical tool, it cannot be detected. Therefore, the proposed algorithm predicts the position of the surgical tool in the current frame using the position of the surgical tool in the previous two frames. This prediction is based on the center point of the surgical tool. More specifically, the motion vector (MV) of the surgical tool is calculated using the position of the surgical tool in the previous two frames (Equation (1)).

$$MV_t = (x_{t-1} - x_{t-2},\ y_{t-1} - y_{t-2}) \tag{1}$$

By adding the value of this motion vector to the position vector of the previous frame ($(x_{t-1},\ y_{t-1})$), the position in the current frame ($(\hat{x}_t, \hat{y}_t)$) is predicted (Equation (2)).

$$(\hat{x}_t, \hat{y}_t) = MV_t + (x_{t-1},\ y_{t-1}) \tag{2}$$

Tool detection is performed again with the use of the trained network by inputting the cropped image at the pre-determined size based on the predicted position of the tool. The size of the newly input image is set to be less than or equal to 416 × 416, which is the size of the input image of YOLO9000, so that the smaller objects can also be visible more easily. Comparison of the second result to the first result obtained based on the tool detection process, and if the intersection of the union (IOU) of the bounding box of the two results is more than 0.5, we regard that the same tool is detected twice. Accordingly, we discard the second result.

3. Experimental Results and Error Analysis

3.1. Experimental Conditions and Results

We performed experiments on Ubuntu 16.04 using a GPU NVIDIA GeForce GTX 1080, with 16 GB of memory, and a CPU Intel core i7–4770K. The training dataset was created with the use of vertical flip, horizontal flip, or both, to generate the 1st to the 7th videos at m2cai16-tool-locations, thus resulting in 7492 images in total (Figure 3). In addition, the 10th video of m2cai16-tool-locations was used as the validation set. Regarding the test set, the 8th and 9th videos from m2cai16-tool-locations [31] and the videos 11–15 of the m2cai16-tool dataset [32] were used. The number of each class and the total number of images included in training and test videos are shown in Table 1.

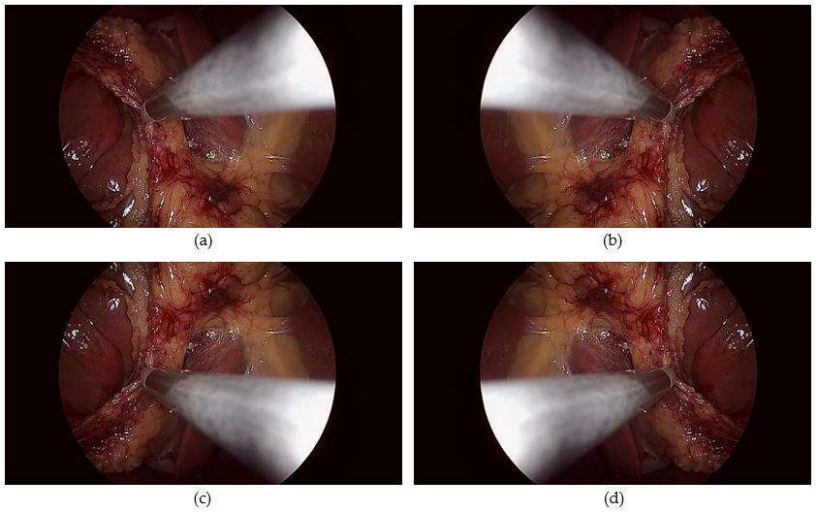


Figure 3. Image augmentation with flip. (**a**) original image, (**b**) horizontal flip, (**c**) vertical flip, (**d**) horizontal and vertical flip.

Table 1. Number of classes and images in training and test datasets.

Dataset (Video Number)	Training	Test
	1~7 (m2cai16-tool-locations)	8~9 (m2cai16-tool-locations) 11~15 (m2cai16-tool)
Grasper	1344	6764
Bipolar	1351	502
Hook	913	7503
Scissors	832	210
Clipper	1106	357
Irrigator	1030	321
SpecimenBag	1505	560
All tools	8081	16,217
Number of images	7492	13,137

Furthermore, in the proposed method, the anchor size of the bounding box uses five pairs of values obtained from the size of the ground-truth box of the training set based on k-means clustering. The value pairs of the five boxes are (15.84, 20.17), (5.37, 7.36), (8.09, 8.98), (8.48, 12.10), and (12.08, 13.37). The weight used in training was pre-trained using visual object classes (VOC), and nonmaximal suppression (NMS) [28] was applied.

We compared the performance of the proposed method with results presented in other studies conducting experiments on the same dataset. Table 2 and Figure 4 show the performance estimates for our proposed algorithm, for the winner of the 2016 M2CAI Tool Presence Detection Challenge, and for the algorithm based on the Faster R-CNN [33]. We also compared the performance of the proposed method—the algorithm using the second version of YOLO and motion vector prediction—with the results obtained in our previous work [34] for the algorithm using the first version of YOLO. Moreover, we performed the comparison of the proposed algorithm with the deformable part models (DPM) [35] and EndoNet [19], which used different datasets to detect surgical tools. The performance comparison was conducted based on the mAP estimates [24]. As shown in Table 2, the proposed method has a higher mAP than the alternative algorithms including the winners of the M2CAI Tool Presence Detection Challenge. This observation was obtained based on the average of all considered tools. Figure 4 shows the mAP values for each class of algorithms, except for the Raju study. The proposed algorithm showed lower performance than some algorithms for such surgical instruments as hook and clipper, but the mAP of all classes was over 80, showing uniform performance regardless of class.

Table 2. Comparison performance using mAP.

Instrument Type	Raju [21]	Sahu [20]	Twin [22]	DPM [35]	EndoNet [19]	A.Jin [24]	B.Choi [34]	Proposed Algorithm
Grasper	NA	73.9	82.2	82.3	84.8	87.2	89.3	92.1
Bipolar	NA	40.8	50.3	60.6	86.9	75.1	32.4	82.3
Hook	NA	95.1	89.4	93.4	95.6	95.3	93.2	85.9
Scissors	NA	26.2	17.0	23.4	58.6	70.8	66.6	81.2
Clipper	NA	35.3	43.6	68.4	80.1	88.4	90.3	85.3
Irrigator	NA	33.2	12.5	40.5	74.4	73.5	42.4	82.9
SpecimenBag	NA	76.6	72.2	40.0	86.8	82.1	91.4	83.2
All tools	63.8	54.5	52.5	58.4	81.0	81.8	72.26	84.7

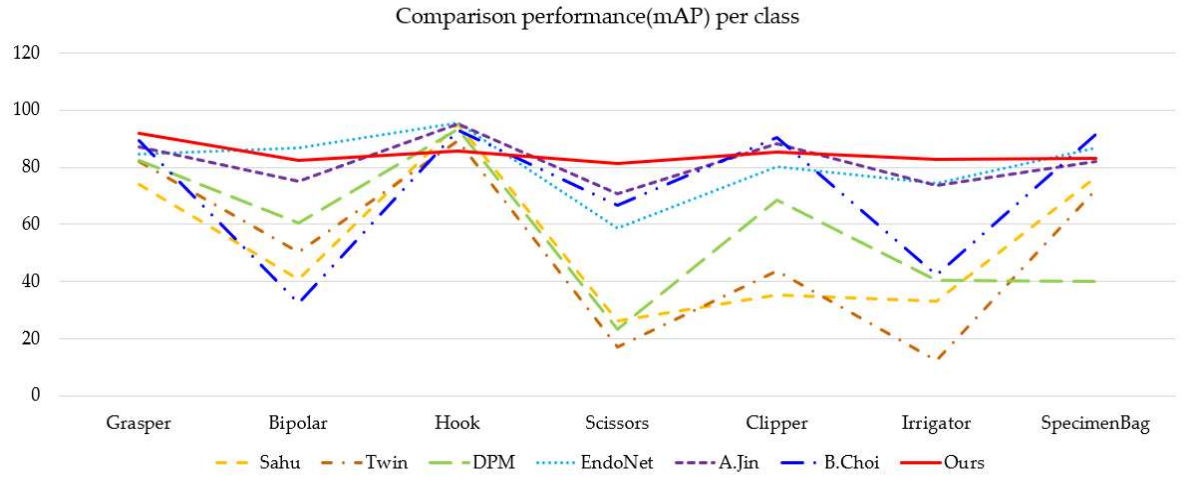


Figure 4. Comparison of mean average precision (mAP) estimates by class for all considered algorithms outlined in Table 2, except the Raju method.

Table 3 compares the speed of the proposed algorithm against that of three different algorithms—two algorithms with high performance according to the results provided in Table 2, and an algorithm using random forests [25]. Algorithms using random forests automatically generate bounding boxes and determine the instrument type of the bounding box. The speed comparison is

based on frames per second (FPS) and allows estimating the accuracy of each algorithm. The accuracy estimate of each algorithm was based on the values provided in corresponding papers, therefore, different criteria were considered. Considering the alternative algorithms with similar average mAP, it can be seen that the proposed algorithm is approximately 7 times faster. Moreover, the proposed algorithm has approximately 1.71 times faster speed and 1.72 times higher accuracy than the random forest algorithm.

Table 3. Comparison of performance of our method with and without missing tool detection.

Algorithm	Bodenstedt [25]	EndoNet [19]	A. Jin [24]	Proposed Algorithm
Accuracy (mAP)	-	81.0	81.8	84.7
Accuracy (%)	86.0	-	-	95.5
Speed (FPS)	22.2	5.0	5.0	38.0

The results of the proposed method are shown in Figure 5. If a tool identified in the previous frame (a) is not found in the current frame (b), the missing tool detection algorithm is applied. (c) is the result of missing tool detection. After the presence of the missing tool is recognized, a white O symbol is displayed in the upper left corner of the image (c, d). Taking Figure 5 as an example, we can describe in more detail that an irrigator is detected in the previous frame (a), however, in the current frame, no surgical tools were detected through YOLO9000 (b). Therefore, through the surgical tool mapping applied on the previous frame and the current frame, it is recognized that the missing tool exists. This is indicated by the white O symbol in the upper left corner of the image. Thereafter, the missing irrigator is detected through the motion vector predicting step, and the class of the detected tool is displayed under the white O symbol.

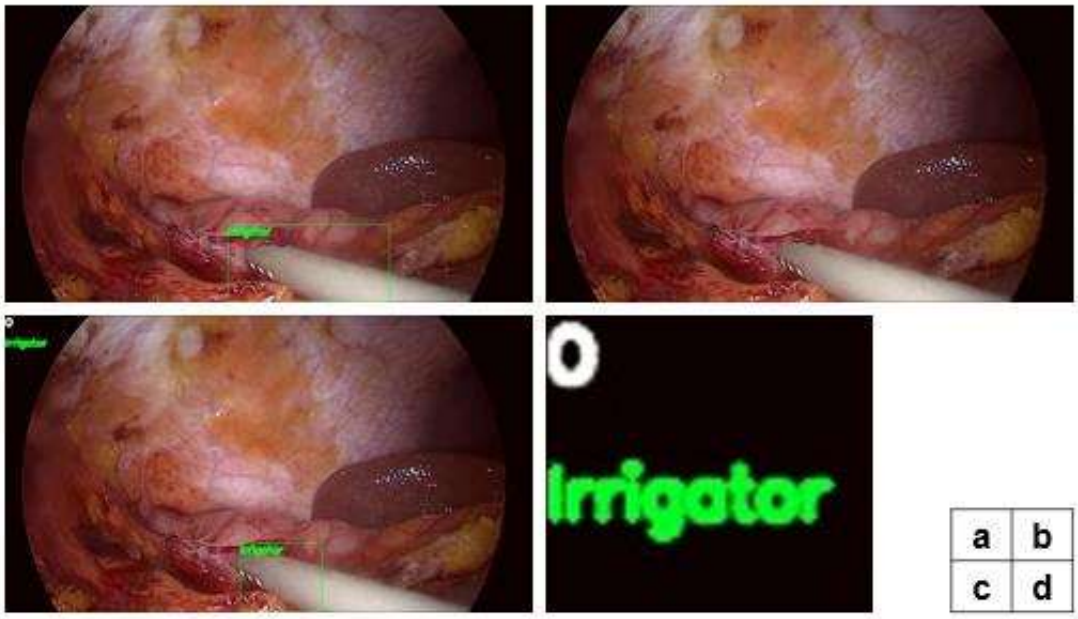


Figure 5. Results of the proposed method. (**a**) The result in the previous frame (t-1). (**b**) As a result of applying only YOLO9000 to the current frame (t), it is found that a missing tool exists. (**c**) YOLO9000 and the missing tool detection algorithm are executed in the current frame (t), and as a result, the missing surgical tool is detected. (**d**) An enlarged view of the upper left corner of image (**c**) (the symbol O in white indicates that the tool is missing, and the class name shown below indicates the class of the tool detected through missing tool detection).

Table 4 compares precision, recall, and F1 scores, according to whether the missing tool detection algorithm is applied or not. Application of missing tool detection allows the precision to be reduced by

approximately 0.63%, the recall by 4.95%, and the F1 score by approximately 2.35%. The reason for the precision decrease is attributed to the erroneous detection of a tool as a missing tool in YOLO9000. Accordingly, an additional detection process is executed.

Table 4. Comparison of performance estimates of the proposed method with and without missing tool detection.

Proposed Method	Precision	Recall	F1 Score
Yolo9000	0.957	0.767	0.851
Yolo9000 + Missing tool detection	0.951	0.805	0.871

3.2. Error Analysis

In the object detection task, errors can be classified as false positive and false negative. A false positive is that the ground truth is false, but the test result is true. In other words, a non-existent surgical tool is detected. For example, the background is erroneously detected as a surgical tool, or the class of the surgical tool is identified incorrectly. A false negative, on the other hand, means that the ground truth is true, but the test result is false. Therefore, it can be concluded that a surgical tool exists, but cannot be detected.

Figure 6 shows false positives and false negatives observed in detecting surgical instruments using only YOLO9000. The above two images are examples of false positives. More specifically, the background was detected as a surgical tool in the upper left image, and a hook was detected incorrectly as a bipolar in the upper-right image. In this case, the nonexistent bipolar is detected, and existing hook is not detected. Consequently, both false positive and false negative are increased by 1. The bottom images are examples of false negatives. The image on the left is an example of failure to detect a grasper, and the image on the right is an example of failure to detect a hook.

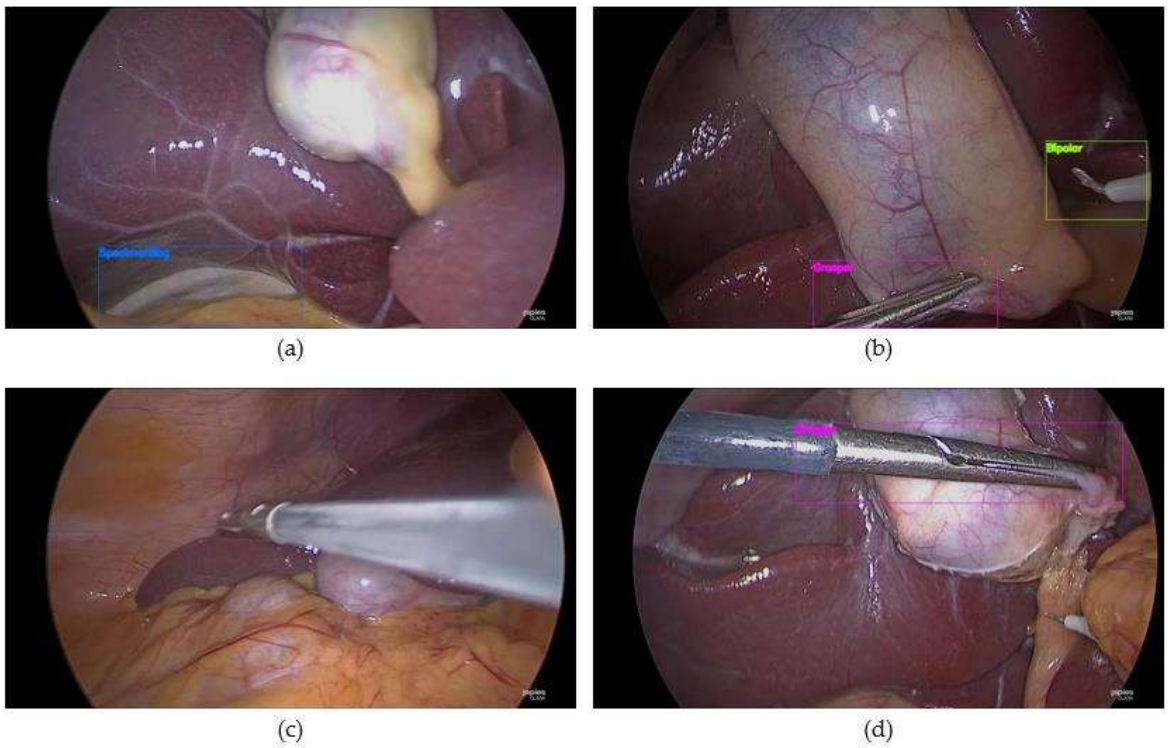


Figure 6. Example of errors in surgical tools detection using YOLO9000 only (**a**,**b**) false positive, (**c**,**d**) false negative.

Figure 7 shows the error in each considered surgical video when using YOLO9000 only, and when using both YOLO9000 and missing tool detection. Each of the six pictures represents an error in each video. However, m2cai16-tool-location videos are displayed together because the total number of frames is small. The numbers on the vertical axis represent the number of errors. For example, if the number of surgical instruments erroneously detected in the same frame is two, the error is also registered as two. The bright blue region of the graph represents a false positive, and the yellow dot region represents a false negative. The orange line indicates the total number of errors. In each figure, the bar on the left shows the error when using only YOLO9000, and that one on the right shows the result obtained using missing tool detection together with YOLO9000.

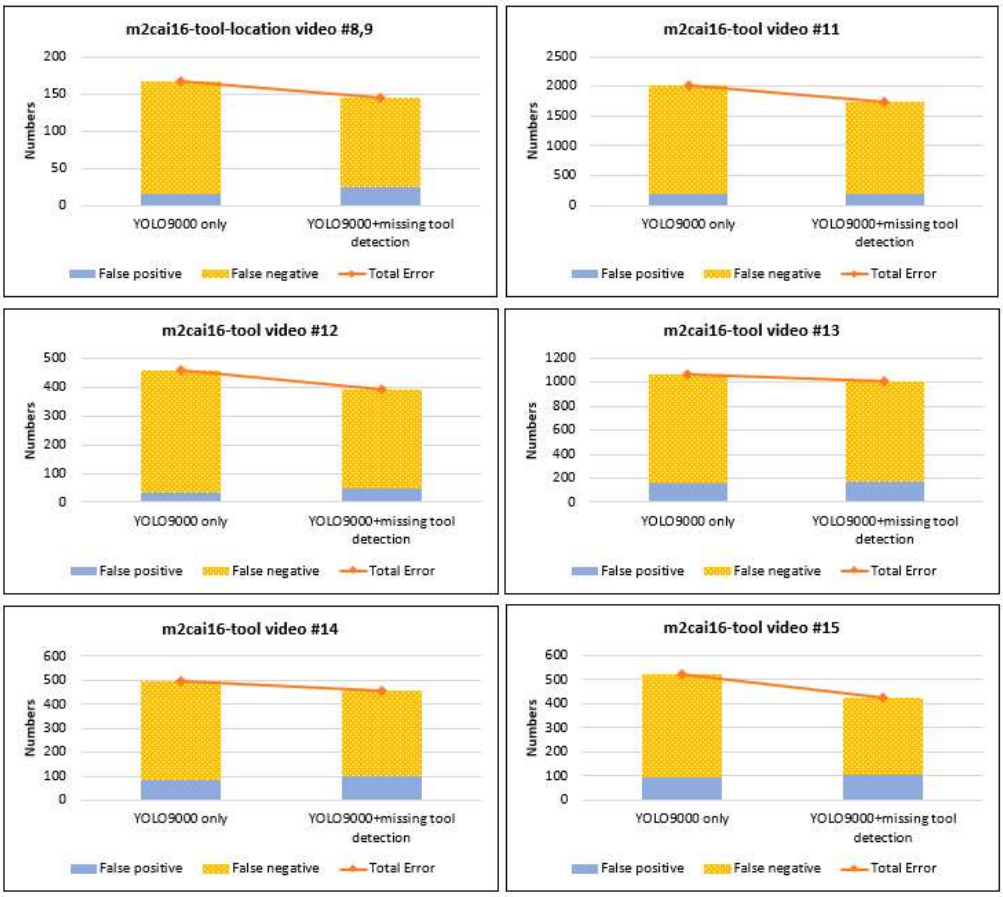
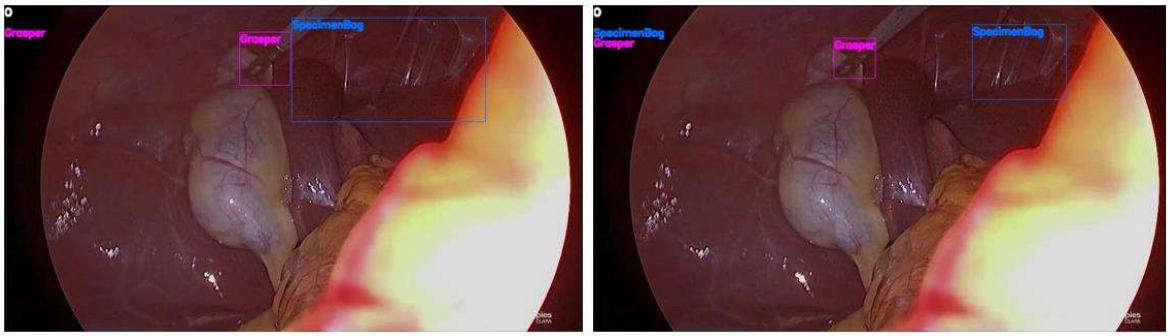


Figure 7. Graph of changes in surgical tool detection error in each test video.

As shown in the figure, when using only YOLO9000, most errors are false negatives. It can be explained by the fact that there are many errors due to missing tools. To solve this problem, we additionally applied the missing tool detection algorithm. As a result, the total number of errors decreased, as shown in the right graph. In addition, the number of false negatives also decreased. On the other hand, the number of false positives increased, because wrongly detected surgical tools were judged to be missing tools and consequently, were redetected accordingly. Figure 8 shows an example of error caused by missing tool detection. The left image is the result obtained in the previous frame. In the previous frame, a grasper was detected correctly through missing tool detection. However, a part of the background was detected as a specimenbag. As a result, in the current frame (right image), the specimenbag was judged as a missing tool through mapping. Correspondingly, the background was detected incorrectly as a specimenbag again due to applying the missing tool detection algorithm.



(**a**) (**b**)

Figure 8. Example of errors in surgical tool detection using YOLO9000 and missing tool detection (**a**) The specimenbag was detected incorrectly in the previous frame, (**b**) The incorrectly detected specimenbag is judged as the missing tool, hence, it is redetected by applying the missing tool detection algorithm in the current frame.

4. Discussion and Conclusions

In this paper, we proposed the new method of detecting and classifying surgical instruments in laparoscopic images. This method has two main advantages—it can be used during real-time operations, and it is robust in comparison to the existing methods.

Firstly, the proposed method can detect surgical tools in real time by using the object detection system YOLO9000. Unlike other methods, You Only Look Once (YOLO) does not allow for finding the region of interest (ROI). Conventional methods aim to identify the ROI from an input image and thereafter, to classify each ROI. However, applying YOLO allowed for the diminishing of the time required to calculate the ROI. YOLO divides an input image into a set of grid cells and then, performs classification of each grid cell. Owing to this key feature of YOLO, the proposed algorithm can detect surgical tools in real time (Table 3).

Moreover, the proposed method is deemed to be robust. In other words, the proposed method demonstrates the uniform and excellent performance in the detection of surgical instruments of all classes. Based on the results provided in Table 2 and Figure 4, it can be concluded that in comparison to other algorithms, the proposed method has a uniform mean average precision (mAP)—over 80—for all classes of surgical instruments, and the highest average mAP with respect to all considered surgical tools. As shown in Figure 4, while the performance of other algorithms with the similar mAP deteriorates for certain classes, the performance of the proposed algorithm is plotted as a flat graph, which confirms its high robustness.

Achieving the robustness of the proposed algorithm is possible owing to the use of the upgraded version of YOLO—YOLO9000. As mentioned earlier, YOLO has a high processing speed, as grid cells are considered instead of ROI. However, it has the problem of lacking accuracy in the first version of YOLO. This can be seen by comparing the performance results of [34] and [24] in Table 2. The study [34] is dedicated to the detection of surgical instruments using the early version of YOLO, and [24] is a study in which surgical instruments were detected applying the faster R-CNN, a typical algorithm using ROI. The results presented in Table 2 and Figure 4 show that the performance of the early version of YOLO is lower than that of the approach based on ROI identification. YOLO9000 has come out to solve these problems. As shown in Figure 2, compared to the earlier version of YOLO, YOLO9000 has subdivided the input image into smaller grid cells resulting in more sophisticated detection.

Another reason for the robustness of the proposed algorithm is that it enables improvements to the detection performance of successive surgical tools owing to the prediction of missing tools. Missing tool detection leads to better performance, as it enables the redetection of surgical tools that have been present in the previous frame, but are not detected in the current frame. As YOLO9000 uniformly divides the input image into grid cells, detection performance may deteriorate when the

main feature is located at the boundary of the grid cell. This situation can occur, for example, as the surgical tool moves. Therefore, it is possible to improve the detection performance by predicting the motion trajectory of the surgical tool and adjusting the position of the grid cell correspondingly. Figure 7 shows the difference in error estimates depending on the presence or absence of the missing tool detection algorithm. The results provided in Table 3 also demonstrate the improved performance owing to this algorithm.

In conclusion, for the purpose of this study we applied two algorithms—YOLO9000 and missing tool detection—to perform the robust detection of surgical instruments in real time. Although the proposed method allows for the diminishing of the error of YOLO9000 by using missing tool detection, the detection error still exists. In particular, missing tool detection requires information from previous frames; therefore, if YOLO9000 detects a surgical tool incorrectly in the previous frame, it consequently affects the current frame.

To solve these problems, it is necessary to use missing tool detection in training. For example, we can obtain a better performance of the proposed method by checking the occurrence of a missing tool in training and adaptively adjusting the probability of the surgical tool presence in the previous frame. Alternatively, a method of using information from previous frames in training through time–sequence techniques such as long short-term memory (LSTM) [36] may be helpful for improving the performance. Finally, increasing the accuracy of the dataset may enable improvements to the detection performance. In this paper, we used an open dataset, which does not reflect information if a surgical tool appears small or obscured; consequently, the detection performance of the proposed method can be improved further if this problem is addressed.

Author Contributions: Project administration, J.C.; Software, Y.C.; Supervision, J.W.C.; Writing—original draft, K.J.

Funding: This research was supported by a grant of the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health and Welfare (grant number: HI17C2410) and a grant (W15-265) from the Asan Institute for Life Sciences, Asan Medical Center, Seoul, Korea.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Weiser, T.G.; Regenbogen, S.E.; Thompson, K.D.; Haynes, A.B.; Lipsitz, S.R.; Berry, W.R.; Gawande, A.A. An estimation of the global volume of surgery: A modelling strategy based on available data. *Lancet* **2008**, *372*, 139–144. [CrossRef]
2. Luo, J.; Yang, C.; Wang, N.; Wang, M. Enhanced teleoperation performance using hybrid control and virtual fixture. *Int. J. Syst. Sci.* **2019**, *50*, 451–462. [CrossRef]
3. Luo, J.; Yang, C.; Su, H.; Liu, C. A Robot Learning Method with Physiological Interface for Teleoperation Systems. *Appl. Sci.* **2019**, *9*, 2099. [CrossRef]
4. Luo, J.; Yang, C.; Li, Q.; Wang, M. A task learning mechanism for the telerobots. *Int. J. Hum. Robot.* **2019**, 1950009. [CrossRef]
5. Trehan, A.; Barnett-Vanes, A.; Carty, M.J.; McCulloch, P.; Maruthappu, M. The impact of feedback of intraoperative technical performance in surgery: A systematic review. *BMJ Open* **2015**, *5*, e006759. [CrossRef] [PubMed]
6. Mansoorian, M.R.; Hosseiny, M.S.; Khosravan, S.; Alami, A.; Alaviani, M. Comparing the effects of objective structured assessment of technical skills (OSATS) and traditional method on learning of students. *Nurs. Midwifery Stud.* **2015**, *4*. [CrossRef] [PubMed]
7. Vassiliou, M.C.; Feldman, L.S.; Andrew, C.G.; Bergman, S.; Leffondré, K.; Stanbridge, D.; Fried, G.M. A global assessment tool for evaluation of intraoperative laparoscopic skills. *Am. J. Surg.* **2005**, *190*, 107–113. [CrossRef] [PubMed]
8. Van Empel, P.J.; van Rijssen, L.B.; Commandeur, J.P.; Verdam, M.G.; Huirne, J.A.; Scheele, F.; Meijerink, W.J. Objective versus subjective assessment of laparoscopic skill. *ISRN Minim. Invasive Surg.* **2013**. [CrossRef]
9. Bouget, D.; Allan, M.; Stoyanov, D.; Jannin, P. Vision-based and marker-less surgical tool detection and tracking: A review of the literature. *Med. Image Anal.* **2017**, *35*, 633–654. [CrossRef] [PubMed]

10. Jo, K.; Choi, B.; Choi, S.; Moon, Y.; Choi, J. Automatic Detection of Hemorrhage and Surgical Instrument in Laparoscopic Surgery Image. In Proceedings of the 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Orlando, FL, USA, 16–20 August 2016; pp. 1260–1263.
11. Cai, K.; Yang, R.; Lin, Q.; Wang, Z. Tracking multiple surgical instruments in a near-infrared optical system. *Comput Assist. Surg.* **2016**, *21*, 46–55. [CrossRef] [PubMed]
12. Kranzfelder, M.; Schneider, A.; Fiolka, A.; Schwan, E.; Gillen, S.; Wilhelm, D.; Feussner, H. Real-time instrument detection in minimally invasive surgery using radiofrequency identification technology. *J. Surg. Res.* **2013**, *185*, 704–710. [CrossRef] [PubMed]
13. Laina, I.; Rieke, N.; Rupprecht, C.; Vizcaíno, J.P.; Eslami, A.; Tombari, F.; Navab, N. Concurrent Segmentation and Localization for Tracking of Surgical Instruments. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Cham, Switzerland, 2017; pp. 664–672.
14. Lee, C.; Wang, Y.F.; Uecker, D.R.; Wang, Y. Image Analysis for Automated Tracking in Robot-Assisted Endoscopic Surgery. In Proceedings of the 12th International Conference on Pattern Recognition, Jerusalem, Israel, 9–13 October 1994; pp. 88–92.
15. Reiter, A.; Allen, P.K. An Online Learning Approach to In-Vivo Tracking Using Synergistic Features. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Taipei, Taiwan, 18–22 October 2010; pp. 3441–3446.
16. Bouget, D.; Benenson, R.; Omran, M.; Riffaud, L.; Schiele, B.; Jannin, P. Detecting surgical tools by modelling local appearance and global shape. *IEEE Trans. Med. Imaging* **2015**, *34*, 2603–2617. [CrossRef] [PubMed]
17. Reiter, A.; Allen, P.K.; Zhao, T. Feature Classification For tracking Articulated Surgical Tools. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Berlin, Germany, 2012; pp. 592–600.
18. Speidel, S.; Benzko, J.; Krappe, S.; Sudra, G.; Azad, P.; Müller-Stich, B.P.; Dillmann, R. Automatic Classification of Minimally Invasive Instruments Based on Endoscopic image Sequences. In *Medical Imaging 2009: Visualization, Image-Guided Procedures, and Modeling*; International Society for Optics and Photonics: Bellingham, WA, USA, 2009; p. 72610A.
19. Twinanda, A.P.; Shehata, S.; Mutter, D.; Marescaux, J.; De Mathelin, M.; Padoy, N. A deep architecture for recognition tasks on laparoscopic videos. *IEEE Trans. Med. Imaging* **2016**, *36*, 86–97. [CrossRef]
20. Sahu, M.; Mukhopadhyay, A.; Szengel, A.; Zachow, S. Tool and phase recognition using contextual CNN features. *arXiv*, 2016; arXiv:1610.08854.
21. Raju, A.; Wang, S.; Huang, J. *M2CAI Surgical tool Detection Challenge Report*; Technical Report; University of Texas at Arlington: Arlington, TX, USA, 2016.
22. Twinanda, A.P.; Mutter, D.; Marescaux, J.; de Mathelin, M.; Padoy, N. Single-and multi-task architectures for tool presence detection challenge at M2CAI 2016. *arXiv* **2016**, arXiv:1610.08851.
23. Tool Presence Detection Challenge Results. Available online: http://camma.u-strasbg.fr/m2cai2016/index.php/tool-presence-detection-challenge-results (accessed on 18 July 2019).
24. Jin, A.; Yeung, S.; Jopling, J.; Krause, J.; Azagury, D.; Milstein, A.; Fei-Fei, L. Tool Detection and Operative Skill Assessment in Surgical Videos Using Region-Based Convolutional Neural Networks. In Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 691–699.
25. Bodenstedt, S.; Ohnemus, A.; Katic, D.; Wekerle, A.L.; Wagner, M.; Kenngott, H.; Speidel, S. Real-time image-based instrument classification for laparoscopic surgery. *arXiv*, 2018; arXiv:1808.00178.
26. Shvets, A.A.; Rakhlin, A.; Kalinin, A.A.; Iglovikov, V.I. Automatic Instrument Segmentation in Robot-Assisted Surgery Using Deep Learning. In Proceedings of the 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; pp. 624–628.
27. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
28. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
29. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. *Proc. Icml* **2013**, *30*, 3.
30. Redmon, J.; Farhadi, A. An incremental improvement. *arXiv*, 2018; arXiv:1804.02767.

31. Available online: http://ai.stanford.edu/~{}syyeung/tooldetection.html (accessed on 3 July 2019).
32. Available online: http://camma.u-strasbg.fr/m2cai2016/index.php/program-challenge (accessed on 3 July 2019).
33. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. In Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS'15), Montreal, QC, Canada, 7–12 Decembar 2015.
34. Choi, B.; Jo, K.; Choi, S. Surgical-Tools Detection Based on Convolutional Neural Network in Laparoscopic Robot-Assisted Surgery. In Proceedings of the 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Jeju Island, Korea, 11–15 July 2017.
35. Felzenszwalb, P.F.; Girshick, R.B.; McAllester, D.; Ramanan, D. Object detection with discriminatively trained part based models. *Trans. PAMI* **2010**, *32*, 1627–1645. [CrossRef] [PubMed]
36. Sak, H.; Senior, A.; Beaufays, F. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling. In Proceedings of the Fifteenth Annual Conference of the International Speech Communication Association, Singapore, 14–18 September 2014.