



HAL
open science

PEg TRAnsfer Workflow recognition challenge report: Do multimodal data improve recognition?

A. Huaulme, K. Harada, Q.-M. Nguyen, B. Park, S Hong, M. K. Choi, M. Peven, Y. Li, Y Long, Q. Dou, et al.

► **To cite this version:**

A. Huaulme, K. Harada, Q.-M. Nguyen, B. Park, S Hong, et al.. PEg TRAnsfer Workflow recognition challenge report: Do multimodal data improve recognition?. *Computer Methods and Programs in Biomedicine*, 2023, 236, pp.107561. 10.1016/j.cmpb.2023.107561 . hal-04089303

HAL Id: hal-04089303

<https://hal.science/hal-04089303>

Submitted on 6 Dec 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Highlights

- Peg Transfer data set containing video, kinematic, semantic segmentation and workflow annotation.
- Challenge of surgical workflow recognition with different modality
- Comparison of multiple deep learning based recognition methods

Journal Pre-proof

PEg TRAnsfer Workflow recognition challenge report: Do multimodal data improve recognition?

Arnaud Huaulmé^{a,*}, Kanako Harada^b, Quang-Minh Nguyen^a, Bogyu Park^c,
Seungbum Hong^c, Min-Kook Choi^c, Michael Peven^d, Yunshuang Li^e, Yonghao
Long^f, Qi Dou^f, Satyadwyoom Kumar^g, Seenivasan Lalithkumar^h, Ren
Hongliang^{h,i}, Hiroki Matsuzaki^j, Yuto Ishikawa^j, Yuriko Harai^j, Satoshi Kondo^k,
Manoru Mitsuishi^b, Pierre Jannin^{a,*}

^aUniv Rennes,INSERM, LTSI - UMR 1099, F35000, Rennes, France

^bDepartment of Mechanical Engineering, the University of Tokyo, Tokyo 113-8656, Japan

^cVisionAI hutom, Seoul, Republic of Korea

^dJohns Hopkins University, Baltimore, USA

^eZhejiang University, Hangzhou, China

^fDepartment of Computer Science & Engineering, The Chinese University of Hong Kong

^gNetaji Subhas University of Technology, Delhi, India

^hNational University of Singapore, Singapore, Singapore.

ⁱThe Chinese University of Hong Kong, Hong Kong, Hong Kong

^jNational Cancer Center Japan East Hospital, Tokyo 104-0045, Japan

^kMuroran Institute of Technology, Hokkaido, Japan

Abstract

Background and Objective: In order to be context-aware, computer-assisted surgical systems require accurate, real-time automatic surgical workflow recognition. In the past several years, surgical video has been the most commonly-used modality for surgical workflow recognition. But with the democratization of robot-assisted surgery, new modalities, such as kinematics, are now accessible. Some previous methods use these new modalities as input for their models, but their added value has rarely been studied. This paper presents the design and results of the “PEg TRAnsfer Workflow recognition” (PETRAW) challenge with the objective of developing surgical workflow recognition methods based on one or more modalities and studying their added value.

Methods: The PETRAW challenge included a data set of 150 peg transfer sequences performed on a virtual simulator. This data set included videos,

*Corresponding author

Email addresses: arnaud.huaulme@univ-rennes1.fr (Arnaud Huaulmé),
pierre.jannin@univ-rennes1.fr (Pierre Jannin)

kinematic data, semantic segmentation data, and annotations, which described the workflow at three levels of granularity: phase, step, and activity. Five tasks were proposed to the participants: three were related to the recognition at all granularities simultaneously using a single modality, and two addressed the recognition using multiple modalities. The mean application-dependent balanced accuracy (AD-Accuracy) was used as an evaluation metric to take into account class balance and is more clinically relevant than a frame-by-frame score.

Results: Seven teams participated in at least one task with four participating in every task. The best results were obtained by combining video and kinematic data (AD-Accuracy of between 93% and 90% for the four teams that participated in all tasks).

Conclusion: The improvement of surgical workflow recognition methods using multiple modalities compared with unimodal methods was significant for all teams. However, the longer execution time required for video/kinematic-based methods (compared to only kinematic-based methods) must be considered. Indeed, one must ask if it is wise to increase computing time by 2,000 to 20,000% only to increase accuracy by 3%. The PETRAW data set is publicly available at www.synapse.org/PETRAW to encourage further research in surgical workflow recognition.

Keywords: Surgical Process Model, Workflow recognition, Multimodal, OR of the future

1. Introduction

To fully integrate computer-assisted surgery systems in the operating room, a complete and explicit understanding of the surgical procedure is needed. A surgical process model (SPM) is a “simplified pattern of a surgical process that reflects a predefined subset of interest of the surgical process in a formal or semi-formal representation” [1], thus allowing for the surgical procedure to be rigorously modeled and described. The SPM methodology consists of decomposing a surgical procedure into five increasingly-coarse levels of granularity: dexeme,

9 surgeme, activity, step, and phase [2, 3]. A dexeme, the lowest granularity level,
10 is a numeric representation of the motion. A surgeme represents a surgical
11 motion with an explicit semantic interpretation of the immediate motion (e.g.,
12 pulling). An activity describes the motion's overall action (action verbs; e.g.,
13 cut) performed on a specific target (e.g., the pouch of Douglas) by a specific
14 surgical instrument (e.g., a scalpel). A step is the succession of these activities
15 which together achieve a specific surgical objective (e.g., resection of the pouch
16 of Douglas). Finally, a phase is the succession of steps that constitute a main
17 period of the intervention (e.g., resection). SPM's are used for learning and
18 expertise assessment [4, 5], robot assistance [6], operating room optimization
19 and management [7, 8], decision-making support [9], and quality supervision
20 [10].

21 The primary limitation of the state-of-the-art in SPM's [3, 4, 5, 7, 9, 10] is their
22 need to be manually interpreted by human observers, which is observer-dependent,
23 time-consuming, and subject to error [11]. Thus, the proposed solutions can
24 not be directly used to bring context-awareness into computer-assisted surgery
25 applications in the operating room. To overcome this limitation, automatic
26 workflow recognition methods have been developed for multiple granularity levels,
27 including phase [8, 12, 13], step [14, 15], and activity [6, 16]. With the emergence
28 of deep learning, most of these recent automatic workflow recognition methods
29 are based on convolutional neural networks, such as AlexNet [17] or ResNet [18];
30 on recurrent neural networks, such as LSTM [19] or gated recurrent unit (GRU)
31 [20]; and more recently on transformers [21].

32 Along with what methodology to use, it is also an open question as to
33 which data modalities should be used as input for this task. In robot-assisted
34 surgery and virtual reality training environments, video and kinematic data are
35 both readily available. Despite this, most state-of-the-art workflow recognition
36 methods are based on a single modality, such as only video [22, 23] or only
37 kinematic data [3, 24]. Few studies have used workflow recognition method based
38 on both video and kinematic data [25, 26, 27]. However, with the exception of
39 the study by Long *et al.*[26], they do not compare the results obtained based on

40 the number and type of input modalities.

41 Semantic segmentation of surgical video is also essential for surgical under-
42 standing and is an active area of research. For example, in five editions of the
43 EndoVis MICCAI Challenge (2015 to 2020), six of the 19 proposed sub-challenges
44 were dedicated to this topic. However, to the best of our knowledge, semantic
45 segmentation has rarely been used as a supplementary task paired with, or as
46 additional input for, surgical workflow recognition.

47 Therefore, the “PEg TRAnsfer Workflow recognition by different modalities”
48 (PETRAW) sub-challenge, which is part of EndoVis, provided a unique data set
49 for automatic recognition of surgical workflows containing video, kinematic, and
50 segmentation data on 150 peg transfer training sequences. Participants were
51 asked to develop model(s) to recognize phases, steps, and activities using one or
52 several of the available modalities.

53 **2. Methods: Challenge Design**

54 This section describes the challenge design, organization, objective, data set,
55 and assessment methods.

56 *2.1. Challenge organization*

57 The PETRAW challenge was a one-time event organized as part of EndoVis
58 during the online 2021 international conference on Medical Image Computing and
59 Computer-Assisted Intervention (MICCAI2021). Four people were involved in the
60 organization: Arnaud Huault and Pierre Jannin from the University of Rennes
61 1 (France), and Kanako Harada and Mamoru Misthuishi from Tokyo University
62 (Japan). Complete information about the challenge was made available to
63 participants using the Synapse platform: www.synapse.org/PETRAW.

64 Challenge participants were subject to the following rules:

- 65 • Participants had to submit a fully automatic method that could recognize
66 phases, steps, and activities on the same model using one or several
67 modalities; and

- 68 • Only data provided by the organizers and publicly available data sets,
69 including pre-trained networks, were authorized for use in training. The
70 publicly available data sets must have been open or otherwise available to
71 all participants at the time the PETRAW data set was released.

72 The results of all participating teams were announced publicly during the chal-
73 lenge day. Challenge organizers and people from the organizing institutions
74 could also participate in the challenge but were excluded from the competitive
75 rankings. Participating teams were encouraged (but not required) to provide
76 their code as open access.

77 For a valid submission, the participating teams had to provide the following
78 elements: a write-up, a Docker image allowing the organizers to compute the
79 results, and a pre-recorded talk to limit technical issues during the challenge day
80 (online event). Multiple Docker images could be submitted, but only the last
81 submission was officially used to generate the evaluation results. No leaderboard
82 or evaluation results were provided prior to the challenge day.

83 The challenge schedule was as follows: The training data set, including videos,
84 kinematic data, and workflow annotations, was released on June 1, 2021; corre-
85 sponding semantic segmentation data was released on June 9, 2021; submissions
86 were accepted until September 12, 2021 (23:59 PST); and the evaluation results
87 were announced on October 1, 2021, during the online MICCAI2021 event. Some
88 teams obtained unexpectedly poor results (i.e., workflow recognition rates inferior
89 to 50%), which made further analysis of the results not relevant. Therefore, each
90 team was allowed to provide a new submission before October 31, 2021. The
91 teams that made a new submission are identified in Section 3.2. The challenge
92 test data set and the organizers' evaluation scripts were released with this paper
93 at www.synapse.org/PETRAW

94 *2.2. Challenge objective*

95 The objective of the PETRAW challenge was to study the contribution of
96 each modality (either alone or in combination) to surgical workflow recognition.
97 To achieve this goal, participants were asked to create a single classification model

98 to determine the surgical task at three levels of granularity (phase, step, and
99 action). Five different tasks were offered as part of the challenge: three concerned
100 the development of unimodal models (i.e., video-based, kinematic-based, or
101 semantic segmentation-based models); and two concerned multimodal-based
102 models. The unimodal-based models were used as a baseline for comparison
103 with the multimodal-based models. In order to keep to a reasonable number of
104 tasks, not all multimodal configurations could be studied. For models based on
105 semantic segmentation data (and to reflect the fact that clinically this modality
106 can be only obtained through a trained segmentation model), participants were
107 asked to use the output of such model as input for PETRAW.

108 *2.3. Challenge data set*

109 The challenge data set was composed of 150 sequences of peg transfer training
110 sessions. The objective of the peg transfer session was to transfer six blocks from
111 the left peg to the right and then back. Each block needed to be extracted from
112 the peg using a grasper (operated by one hand), transferred to the other grasper
113 (in the other hand), and finally inserted onto the peg on the opposite side of the
114 board.

115 All sequences were acquired by a non-medical expert at the LTSI Laboratory,
116 University of Rennes 1, France. The data set was divided into training data
117 (n=90 sequences) and test data (n=60 sequences). Each sequence included
118 kinematic data, video, semantic segmentation of the video for each frame, and
119 workflow annotations at each level of granularity. Only the training data set was
120 provided to participants.

121 *2.3.1. Data acquisition*

122 The challenge data was acquired on a virtual reality simulator (Figure 1)
123 developed at the Department of Mechanical Engineering, University of Tokyo,
124 Japan [28], consisting of a laptop (i7-700HQ, 16Go RAM, GTX 1070), a 3D
125 rendering setup (3D screen: 24 inches, 144Hz; and 3D glasses), and two haptic
126 user interfaces (3D system TouchTM).



Figure 1: The virtual reality simulator used for data acquisition.

127 For data acquisition, a single operator performed a series of five consecutive
128 peg transfer tasks followed by a break of at least 5 hours to limit fatigue. This
129 was repeated 30 times to yield a total of 150 peg transfer task sequences. The
130 COVID-19 crisis (acquisition made in 2020-2021) did not allow us to recruit
131 multiple participants. To limit the effect of immediate learning or fatigue in a
132 single session, three sequences from each series were randomly chosen for training,
133 and the remaining two for testing.

134 The kinematic data and videos were synchronously acquired at 30 Hz during
135 each peg transfer task. Each video had a resolution of 1920x1080 pixels and
136 semantic segmentation was performed for each frame off-line following the task.
137 Kinematic data included the position, rotation quaternion, forceps aperture
138 angle, linear velocity (obtained from simulation, not derived from position), and
139 angular velocity (obtained from simulation, not derived from orientation) of the
140 left and right instruments (i.e., graspers). The position and linear velocity were
141 measured in centimeters and centimeters per second, respectively. The angle and
142 angular velocity were measured in degrees and degrees per second, respectively.

143 The semantic segmentation included six classes (shown in Figure 2): back-

144 ground (black, hexadecimal code:#000000), base (white, #FFFFFF), left instru-
 145 ment (red, #FF0000), right instrument (green, #00FF00), pegs (blue, #0000FF),
 146 and blocks (magenta, #FF00FF).

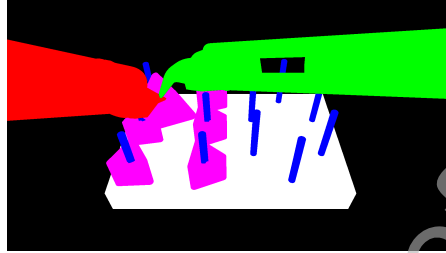


Figure 2: Representative segmentation mask with the six classes: background (black), base (white), left instrument (red), right instrument (green), pegs (blue) and blocks (magenta).

147 The workflow annotations were automatically computed using the scene in-
 148 formation and the ASURA method [11]. The challenge organizers had previously
 149 demonstrated in [11] that ASURA is more accurate and robust than manual
 150 annotation on peg transfer tasks. Two phases, twelve steps, six action verbs, two
 151 targets, and one surgical instrument were identified to describe the workflow
 152 (Table 1). Each phase corresponded to the transfer of all of the blocks in one
 153 direction (e.g. “L2R” for left to right). Each step (six per phase) corresponded to
 154 the transfer of a single block (e.g.“Block1 L2R” for the transfer of the first block
 155 from the left to the right). For the activities, two targets were differentiated:
 156 “block” and “other block”. “Block” corresponds to the one that is currently being
 157 transferred. “Other block” is an additional target used to differentiate when the
 158 user accidentally interacts with any block other than the one to be transferred.

159 One limitation of the method presented by [11] was the inability to accurately
 160 differentiate between the action verbs “catch” and “touch”, as each tool tip was
 161 considered as a unique virtual object. The virtual reality simulator was updated
 162 to include four separating regions rather than one, allowing these actions to
 163 be readily differentiated. Accordingly, the workflow annotations were manually
 164 examined and corrected to ensure annotation quality.

Table 1: Peg-transfer vocabulary.

| Phases | Steps | Activities | | |
|---------------------------------|-------------|------------|-------------|---------|
| | | Verb | Target | Tool |
| Transfer Left To Right (L2R) | Block 1 L2R | Catch | Block | Grasper |
| | Block 2 L2R | Drop | Other block | |
| | Block 3 L2R | Extract | | |
| | Block 4 L2R | Hold | | |
| | Block 5 L2R | Insert | | |
| | Block 6 L2R | Touch | | |
| Transfer Right To Left (R2L) | Block 1 R2L | | | |
| | Block 2 R2L | | | |
| | Block 3 R2L | | | |
| | Block 4 R2L | | | |
| | Block 5 R2L | | | |
| | Block 6 R2L | | | |

165 2.3.2. Data pre-processing

166 The original workflow annotations were formatted in terms of start and finish
 167 time, expressed in milliseconds. These annotations were sampled to provide a
 168 discrete sequence at 30Hz, synchronized with the kinematic, video, and segmenta-
 169 tion data to allow for frame-by-frame annotation. Due to their lack of variability,
 170 the two targets and the tool were not included in the workflow annotation.
 171 Furthermore, when no phase, step, or activity occurred, the term “idle” was used.
 172 For each timestamp, the following information was provided: timestamp_number,
 173 phase_value, step_value, verb_Left_Hand, verb_Right_Hand.

174 2.3.3. Ground truth uncertainties

175 The semantic segmentations were the primary source of uncertainty in the
 176 ground truth. Due to the transformation of 3D meshes into 2D images, some
 177 pixels were attributed to the wrong class, especially at boundaries between the



Figure 3: Zoom of 219x123 pixels from Figure 2 to highlight segmentation errors. Right instrument/block (green/magenta) and left/right instruments (red/green) errors are shown where pixels are labeled as background (black). On this zoom, only 51 pixels were mis-segmented (around 0.2%).

178 right instrument/peg, left instrument/peg, left instrument/block, and left/right
 179 instruments (Figure 3). We estimated this uncertainty by counting the number
 180 of mis-segmented pixels on 10 images that included many boundary regions, such
 181 as those between surgical instruments, pegs, and blocks. On each image, the
 182 number of mis-segmented pixels represents less than 0.25% of the total image.
 183 To take into account the fact that this manual assessment was not representative
 184 of the whole data set, we estimated that this mis-segmentation represents less
 185 than 0.5% of pixels.

186 Workflow annotations were another source of uncertainty. Although the
 187 ASURA method is consistent (i.e., it generates the same result in two identical
 188 situations) and a manual check was performed to limit inaccuracies, some
 189 components could not be recognized with complete certainty. Two particular
 190 instances were identified. First, in sequence 130 of the training data set, the
 191 block in step “Block 1 R2L” was inserted in a non-standard way. Specifically,
 192 the block was released by the operator, and while falling became inserted in
 193 the peg. Therefore, the insert action was absent. The other instance concerned
 194 sequence 79 of the test data set. This time, the operator caught a block before
 195 the previous one had been fully inserted, leading to an overlap between the steps
 196 “Block 5 R2L” and “Block 6 R2L”. The second was chosen as the sole annotation
 197 to maintain the true beginning of the step.

198 2.3.4. Data set characteristics

199 The training and test data sets presented similar characteristics. The mean
200 and standard deviation duration was 140.2 ± 18.9 seconds for the training data set
201 and 141.7 ± 18.0 seconds for the test data set. Figure 4 presents the distribution
202 of every vocabulary component for each granularity level in the training data
203 set (Figures 4a, 4c, 4e, 4g) and the test data set (Figures 4b, 4d, 4f, 4h). Even
204 for underrepresented components, the distribution was very similar in both
205 data sets. For instance, the verb “touch” (left hand) represented 0.59% and
206 0.60% of the samples in the training and test data sets, respectively, and “touch”
207 (right hand) represented 0.62% and 0.48%, respectively. The distribution of
208 each vocabulary component between each data set is only statistically different
209 (Mann-Whitney test) for two steps: “Block 1 L2R” and “Block 6 L2R”, with
210 $p=0.045$ and $p=0.036$ respectively.

211 Another important characteristic of the data sets was the high class unbalance
212 of at least one vocabulary term for each granularity level. For the phases, the
213 term “idle” represented less than 4% of all data, whereas the other phase terms
214 accounted for more than 47% (L2R and R2L). For the steps, the term “idle”
215 represented less than 4%, whereas the non-idle steps accounted for approximately
216 more than 7.5% of each data set (Figures 4a-4d). This unbalance was more
217 pronounced at the action level, where the least represented verb (i.e., “touch”)
218 represented approximately 0.6% of the data set, whereas the verb “idle” accounted
219 for more than 53%. The detailed distribution values for each granularity level in
220 both data sets are provided in supplementary material.

221 2.4. Assessment method

222 2.4.1. Metrics

223 To assess the participants’ workflow recognition models and to take into
224 account the high class unbalance, balanced versions of accuracy, precision, recall,
225 and F1 were used.

226 In practice, however, some small variations in surgical task recognition are
227 not clinically meaningful and do not constitute a true error. Motivated by this,

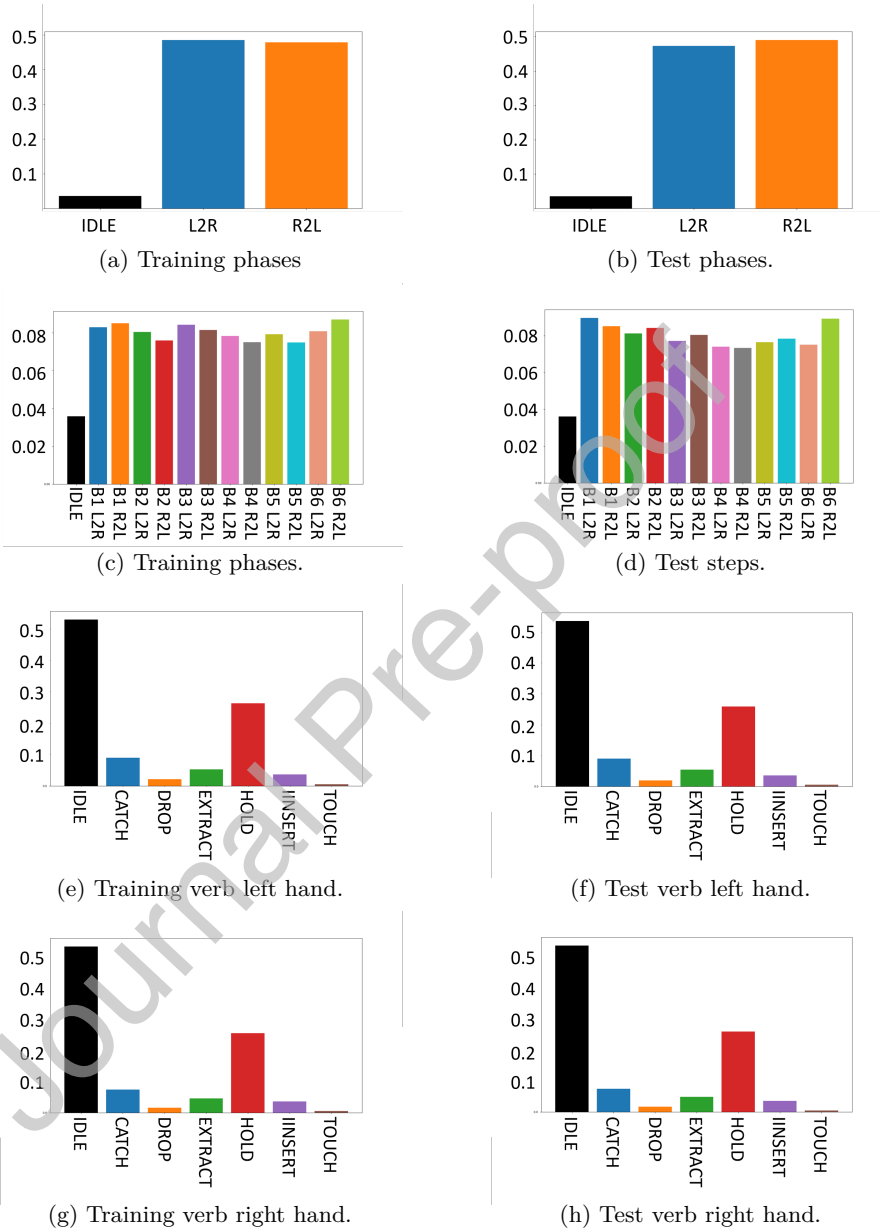


Figure 4: Distribution of each term at each granularity level in the training and test data sets. The y-axis represents the percentage of frames. In (a) and (b), “L2R” means transfer left to right and “R2L” means transfer right to left. In (c) and (d), “B1 L2R” means block 1 left to right, “B2 L2R” means block 2 left to right.

228 Dergachyova *et al.* [29] proposed a re-estimation of these classic frame-by-frame
 229 scores, called application-dependent scores, to take into account an acceptable
 230 delay d . When a predicted transition occurs within a transition window ($2d$)
 231 centered on the ground truth transition, all frames between the two transitions
 232 are considered correct if it is the same transition type (e.g. transition for
 233 verb “catch” or verb “extract”). Therefore, the balanced application-dependent
 234 accuracy (AD-Accuracy) was used and the acceptable delay was fixed at 250 ms.

235 To assess the participants’ segmentation models, the mean Intersection-Over-
 236 Union (IoU) over all classes was also used, also known as the Mean Jaccard
 237 Index over all classes. The IoU is the area of overlap between the predicted
 238 segmentation ($Pred$) and the ground truth (GT), divided by the area of union
 239 between the $Pred$ and the GT . In our cases, there was a multi-class segmentation
 240 problem, therefore the mean IoU value of the image was calculated by taking
 241 the IoU of each class and averaging it over the classes:

$$\begin{aligned}
 MeanIoU_{frame} &= \frac{1}{6} \sum_{class} IoU_{class} \\
 &= \frac{1}{6} \sum_{class} \frac{|GT \cap Pred|_{class}}{|GT \cup Pred|_{class}} \\
 &= \frac{1}{6} \sum_{class} \frac{TP_{class}}{TP_{class} + FP_{class} + FN_{class}},
 \end{aligned} \tag{1}$$

242 where TP (True Positives) is the number of pixels inside the GT area that are
 243 correctly predicted, FP (False Positives) is the number of pixels outside the GT
 244 area but predicted as belonging to the class, and FN (False Negatives) is the
 245 number of pixels inside the GT area that are incorrectly predicted.

246 2.4.2. Ranking method

247 The ranking of the participating methods used only the surgical task recog-
 248 nition metrics. Metrics computed for evaluating the segmentation models were
 249 provided for information purposes only.

250 A metric-based aggregation method using the AD-Accuracy values across
 251 all test sequences was used for the ranking. Metric-based aggregation was used

252 according to the recommendations made in [30], which show it to be one of
 253 the most robust. As all tasks consisted of recognizing the phase, step, and the
 254 actions of the left and right hands (i.e., the left and right verbs), the ranking
 255 score for the algorithm a_i was computed as follows:

$$s(a_i) = \frac{s_{phase}(a_i) + s_{step}(a_i) + s_{verb_left}(a_i) + s_{verb_right}(a_i)}{4} \quad (2)$$

256 with,

$$s_{phase}(a_i) = \frac{\sum_{t=0}^T phase_balance_accuracy_case_t}{T}, \quad (3)$$

257 where T is the number of sequences to test. Similar equations were used for
 258 the other terms ($s_{step}(a_i)$, $s_{verb_left}(a_i)$ and $s_{verb_right}(a_i)$) with a numerator
 259 specific to each, i.e., $\sum_{t=0}^T step_balance_accuracy_case_t$ for $s_{step}(a_i)$, etc.

260 If a participant method did not produce a prediction for one or several
 261 granularity levels, the accuracy given for each missing granularity level was that
 262 expected for uniformly random predictions. For example, if a model did not
 263 predict the phase, s_{phase} would be set to 1/3 corresponding to the phase having
 264 3 potential values. In practice, this was not encountered and each evaluated
 265 model produced results for each level of granularity.

266 Ranking stability was assessed by testing different ranking methods: mean-
 267 ThenRank, medianThenRank, rankThenMean, rankThenMedian, and testBased.
 268 MeanThenRank was chosen for the ranking. MedianThenRank differs from the
 269 previous method because it used the median instead of the mean in equation
 270 3. For rankThenMean and rankThenMedian, first, the results of each sequence
 271 were ranked among participants, and then the final results were the mean or
 272 median of all ranks. The testBased method is based on bootstrapping. The
 273 ranking was considered stable if a team was ranked in the same position with
 274 the majority of ranking methods. If the ranking was not stable according to the
 275 chosen methods, a tie between teams was pronounced. The ranking computation
 276 and analysis were performed with the ChallengeR package provided by [31].

277 2.4.3. Online recognition compatibility

278 To be online compatible, the proposed methods must satisfy two conditions:

- 279 • to produce predictions faster than the duration between the two samples
280 (i.e., faster than 30 Hz); and
- 281 • to be causal (i.e., not use data from a future time point to make predictions).

282 The computation time was not studied because it could not be assessed fairly for
283 all teams. Indeed, the teams provided a unique Docker image for all tasks, and
284 some teams did not write the output file to standard output as it was received,
285 which did not allow for their durations to be precisely measured.

286 To verify that the methods were causal, the online availability of the frames
287 was mimicked. One additional sequence of 10 seconds, corresponding to the
288 transfer of the first block from the left to the right, was recorded. This sequence
289 was used to generate 300 sub-sequences, each one a frame longer than the
290 previous. Thus, the first sequence only contained the information of the first
291 frame, the second one contained the information of the two first frames, etc.
292 The models were run on the 300 sub-sequences and the last prediction of each
293 sub-sequence to create a definitely causal prediction sequence. A method was
294 considered causal if and only if this definitely causal prediction sequence was
295 identical to the prediction sequence given by the full 300 frames. This causality-
296 testing method is fully automated and also takes into account the complete
297 pipeline used to perform the prediction, such as pre- and post-processing steps,
298 which could lead to a non-causal method even if the network only uses causal
299 components. For reasons of computation time and environmental responsibility,
300 this test was not performed on a whole sequence or the whole test data set. By
301 testing the entire data set, we could be more confident in the causality of the
302 proposed methods, but this would quickly display diminishing returns.

303 *2.5. Additional analyses*

304 To further analyze the impact of using multimodal instead of unimodal
305 models, we performed two additional analyses that were not initially included
306 in the challenge design: the statistical significance to use multimodal models
307 instead of unimodal models, and the execution time. These additional analyses

308 only concerned the teams that participated in the multimodal tasks (4 and 5)
309 with a combination of the same or similar models used for the unimodal tasks.

310 *2.5.1. Comparison between unimodal and multimodal models*

311 To assess the impact of each modality and its combinations on automatic
312 workflow recognition, we performed a statistical analysis with the Wilcoxon test.
313 The difference was significant if the p-value was inferior to 0.05.

314 *2.5.2. Execution time*

315 Performance is not the only important factor when developing automatic
316 recognition models. Indeed, environmental aspects must also be taken into
317 account [32]. To answer this question, we examined the execution time to
318 compute the results of the 60 test sequences. These durations were interpolations
319 that assumed the predictions in each task were computed independently and
320 not the real execution time. Indeed, one team (Hutom, see section 3.2.1) used
321 the predictions from tasks 1 to 3 as input for those of tasks 4 and 5, so the
322 interpolation for the multimodal tasks took into account the execution time for
323 the unimodal ones.

324 **3. Results: Reporting of the Challenge Outcomes**

325 *3.1. Challenge submission*

326 By September 12, 2021, 29 participants had registered for the PETRAW
327 challenge: 17 were members of one of the six competing teams. The organizers
328 also submitted results as a non-competing team to provide a baseline. As
329 explained in Section 2.1, some teams obtained unexpected results and three
330 teams resubmitted results for at least one task.

331 *3.2. Information on the participating teams and their methods*

332 This section describes each team, the methods they used, and the tasks in
333 which they participated. Competing teams are presented in alphabetical order
334 and not in terms of their ranking.

335 3.2.1. *Hutom*

336 The Hutom team (Bogyu Park, Seungbum Hong, and Minkook Choi from
337 VisionAI hutom) participated in all proposed tasks. They resubmitted a Docker
338 image for all tasks except the kinematic-based recognition task.

339 Before training, they performed a simple pre-processing step. To preserve
340 temporal information, they split data into clips of 8 frames. They normalized
341 kinematic data by standardizing the raw input without data augmentation. They
342 resized video data to 256×256 pixels, followed by random cropping (224×224
343 pixels) and normalization. The cropping was limited to preserve the spatial
344 information in each frame of the clip. They resized segmentation data to 512×512
345 pixels.

346 They used a similar baseline architecture for tasks based on the same modality.
347 They computed segmentation data from the video recording using a DeepLabV3+
348 architecture [33]. They used a 3D ResNet network [34] for workflow recognition
349 based on the video modality. For the segmentation modality, they used a
350 SlowFast50 network [35] for segmentation-based recognition and a 3D ResNet
351 network for video/kinematic/segmentation-based workflow recognition. They
352 inputted kinematic data on a bi-directional long short-term memory (Bi-LSTM)
353 network [36]. For multimodal recognition tasks, they used a convolutional
354 feature fusion layer to efficiently perform the fusion of the feature output of each
355 modality. They obtained embedding features with individual modal inputs from
356 each model trained accordingly. Then, they compared the embedding features of
357 each modality with those of other modalities to learn the different representations
358 of each modality. They used the stop gradient-based SimSiam method [37] to
359 compare representations between embedding features. Concomitantly, they
360 stacked embedding features by modality into one block as a chunk and fused
361 them into one embedding through a convolution operation. The approach
362 assumed that feature elements for each modality in the same column have similar
363 temporal information in similar positions. For all networks, they used the Adam
364 optimizer and an initial learning rate of $1e^{-3}$, with a combination of Equalization

365 loss v2 [38] and Normsoftmax Loss [39] as long-tail recognition for addressing
366 data imbalance.

367 3.2.2. JHU-CIRL

368 The JHU-CIRL team (Michael Peven and Gregory D. Hager; Johns Hopkins
369 University) participated in the kinematic-based workflow recognition task.

370 They performed an under-sampling of the kinematic data to reduce the time
371 dimension size in order to prevent vanishing gradient issues during training.
372 For the test, they used the same under-sampling. The JHU-CIRL team did
373 not perform any other pre-processing because they considered that besides the
374 positional data, the addition of velocity data was sufficient for the recognition.

375 They used a unidirectional LSTM network [40] to recognize the four workflow
376 components. They trained the model using traditional cross-entropy loss and the
377 Adam optimizer. They paid special attention to the selection of the following
378 hyperparameters: sampling rate, learning rate, LSTM hidden dimension size,
379 and the number of layers in the LSTM. They ran 5-fold cross-validation to obtain
380 results from each of these hyperparameters. Then, they selected the best set of
381 hyperparameters for the final training: 15Hz sampling rate, $1e^{-3}$ learning rate,
382 256 LSTM Hidden dimension, and 2 LSTM layers.

383 3.2.3. MedAIR

384 The MedAIR team (Yunshuang Li, Yonghao Long, and Qi Dou, Zhejiang
385 University and the Chinese University of Hong Kong) participated in three tasks:
386 video-based, kinematic-based, and video/kinematic-based workflow recognition.
387 They resubmitted a Docker image for the video-based workflow recognition task.

388 The MedAIR team resized videos to 224×224 pixels and then augmented the
389 data using a random horizontal flip and a random rotation of 5° . For kinematic
390 data, they used a linear layer to obtain 2048 dimensions from the 28 dimensions
391 to enrich the information.

392 For unimodal-based workflow recognition (video-based and kinematic-based
393 tasks), the MedAIR team used a Trans-SVNet model [41]. First, they trained

394 two different convolutional neural networks (CNN) to extract spatial features,
 395 one for steps and another for left and right verbs. Then, they trained three
 396 multi-stage temporal convolutional networks (TCN) to obtain temporal features
 397 for steps and verbs. Finally, they used three transformer layers to combine
 398 spatial and temporal features to obtain the final output for the three labels.
 399 Phases were not directly predicted by the networks, but identified based on the
 400 predicted step. They used a stochastic gradient descent (SGD) optimizer with a
 401 cross-entropy loss and a learning rate of $5e^{-4}$.

402 For multimodal-based workflow recognition (video/kinematic-based task),
 403 they used a multi-modal relational graph network (MRG-Net) [26]. Like for
 404 unimodal-based workflow recognition, they used two CNNs to extract features
 405 from each frame in the video for steps and verbs. Then, they obtained the step
 406 labels using the original MRG-Net structure, which was the result of the fully
 407 connected layer with the output of three nodes in the graph. For the verb labels,
 408 the MedAIR team used fully connected layers to produce outputs k_t^l and k_t^r , the
 409 final label prediction for left and right verb labels. They identified phases based
 410 on the predicted step. They used an Adam optimizer with cross-entropy loss
 411 and learning rate of $1e^{-4}$.

412 3.2.4. MMLAB

413 The MMLAB team was composed of Satyadwyoom Kumar, Lalithkumar
 414 Seenivasan, and Hongliang Ren from the Netaji Subhas University of Technology,
 415 National University of Singapore, and the Chinese University of Hong Kong.
 416 They participated in the video/kinematic-based recognition task.

417 MMLAB team proposed a multi-task learning model to perform the recogni-
 418 tion. First, each video frame was resized to 224×224 pixels. A ResNet 50 [18]
 419 pre-trained on ImageNet was used to extract visual features for each video frame.
 420 These features were passed with the frame-specific kinematic data through four
 421 label-specific networks (one per component). Each label-specific network was
 422 composed of two LSTMs [19], one for each modality, to capture the temporal
 423 features. The sequential length was set to 5, allowing the model to infer based

424 on the current and past 4 temporal information sets. The resulting temporal
425 features were then passed through a single linear layer for recognition. Each
426 label-specific network was trained independently with cross-entropy loss, Adam
427 optimizer, and a learning rate of $1e^{-3}$ for phase and step recognition, and $1e^{-2}$
428 for hand verbs.

429 3.2.5. NCC NEXT

430 The NCC NEXT team (Hiroki Matsuzaki, Yuto Ishikawa, Kazuyuki Hayashi,
431 Yuriko Harai, and Nobuyoshi Takeshita, National Cancer Center Japan East
432 Hospital) participated in all proposed tasks. They resubmitted a Docker image
433 for all tasks except the kinematic-based recognition task.

434 They resized the initial video frames to a resolution of 512×256 pixels for
435 video-based workflow recognition and of 480×270 pixels for segmentation-based
436 workflow recognition. This was followed by normalization. They did not perform
437 any preprocessing of kinematic data.

438 For video-based workflow recognition they used Xception networks [42] pre-
439 trained on ImageNet, one per component. They used the Radam optimizer [43]
440 with different learning rates with a batch size of 4, $1e^{-3}$ for phases and steps, and
441 $1e^{-4}$ with a cosine decay scheduler for hand verbs. They also used cross-entropy
442 loss.

443 For kinematic-based workflow recognition, the NCC NEXT used the light
444 gradient boosting machine (LightGBM) framework [44]. Like for the previous
445 task, they did the training and tuning of hyperparameters (i.e., learning rate,
446 minimum data in leaf, number of iterations, and number of leaves) separately
447 for each component (Table 2). They chose gradient boosting as a predictor
448 optimizer and the mean absolute error (MAE) as loss of function.

| Parameters | Phase | Step | Verb_Left | Verb_Right |
|------------------|-------|------|-----------|------------|
| Learning rate | 0.1 | 0.05 | 0.05 | 0.05 |
| min_data_in leaf | 9 | 9 | 3 | 9 |
| num_iteration | 200 | 100 | 100 | 50 |
| num_leaves | 11 | 31 | 11 | 11 |

Table 2: Hyperparameters for the kinematic based model developed by the NCC NEXT team

449 The segmentation was performed by a Deeplabv3+ architecture [33] with an
 450 Xception backbone pre-trained on the Pascal visual object classes (PascalVOC)
 451 data set [45]. With the predicted segmentation, they trained a multi-output
 452 classification model, based on the EfficientNetB7 architecture [46], with Radam
 453 optimizer, cross-entropy loss function, a learning rate of 0.0001 with a cosine
 454 decay scheduler, and a batch size of 16.

455 For the multimodal workflow recognition tasks, the NCC NEXT team se-
 456 lected the method used in the three previous tasks that displayed the highest
 457 accuracy for each component. Specifically, for video/kinematic-based workflow
 458 recognition task, they used the video-based architecture for phase and step
 459 recognition and the kinematic-based architecture for hand verb recognition.
 460 For the video/kinematic/segmentation-based model, they used the video-based
 461 architecture for phase recognition, the segmentation-based architecture for step
 462 recognition, and the kinematic-based architecture for hand verb recognition.

463 3.2.6. SK

464 The SK team (Satoshi Kondo, Muroran Institute of Technology) participated
 465 in all proposed tasks.

466 For preprocessing, the SK team resized the images to 640×353 pixels and
 467 then used random shifting (maximum shift size of 10% of the image size), scaling
 468 (0.9 to 1.1 times), rotation (-5 to 5 degrees), color jitter (-0.9 to 1.1 times for
 469 brightness, contrast, saturation, and hue), and Gaussian blurring (maximum
 470 sigma value = 1.0) for data augmentation. Finally, the images were normalized

471 and the kinematic data were normalized in each dimension.

472 For the video-based workflow recognition task, the SK team used an 18-layer
473 ResNet network [18], pre-trained on ImageNet. The SK team omitted the final
474 fully-connected layer of ResNet and fed its input 512-dimensional feature vector
475 into two fully-connected layers to obtain a prediction of the step and hand verbs.
476 Between these fully-connected layers, they inserted one ReLU and Dropout
477 layers. The team used an Adam optimizer, with learning rate changes with
478 cosine annealing with an initial value of $7.2e^{-4}$, and a batch size of 96. The team
479 optimized the initial learning rates for each task with the Optuna library [47].
480 The team chose cross-entropy loss as the loss function, with weights for each
481 class depending on the class frequency for hand verbs. Phases were not directly
482 predicted from the image, but identified based on the predicted step.

483 The SK team used a stacked LSTM [19] with two layers and 28 hidden
484 layers for the kinematic-based workflow recognition task. The LSTM output
485 was fed into three fully connected layers as done for the previous task. The same
486 optimizer and loss function were used. The initial learning rate was $1.5e^{-3}$ with
487 a batch size of 6 and the number of data in a sequence was 30.

488 Image segmentation was done using the U-Net architecture [48] with ResNet18
489 as encoder with the summation of cross-entropy loss and dice loss. The SK team
490 exploited the same model used for the video-based workflow recognition task and
491 for the segmentation-based task. Both models were trained separately with an
492 Adam optimizer and an initial learning rate of $2.4e^{-5}$ with a batch size of 32 for
493 segmentation, and a learning rate of $1e^{-4}$ with a batch size of 6 for recognition.

494 For the video/kinematic-based task and video/kinematic/segmentation-based
495 task, the SK team ensembled the previously trained dedicated modality networks
496 to obtain a new prediction. As the SK team used the network parameters trained
497 for the previous task, they did not train any network for these tasks.

498 *3.2.7. MediCIS: non-competing team*

499 The MediCIS team was a non-competing team due to the presence of challenge
500 organizers (Quang-Minh Nguyen and Arnaud Huaultmé, University of Rennes 1).

501 The team participated in all proposed tasks.

502 For the preprocessing step, they resized the frames to 256×512 pixels.
 503 Additionally, to train the segmentation model, they down-sampled the data to 6
 504 Hz. They z-normalized the kinematic data.

505 For the video-based workflow recognition task, the MediCIS team used a
 506 hierarchical ResNet50 network [18] pre-trained on ImageNet to extract spatial
 507 features. Then, they used a Multi-Stage Temporal Convolutional Network called
 508 MS-TCN++ [49], with two stages, trained from scratch.

509 For the kinematic-based workflow recognition task, they directly used data
 510 as features for a two-stage MS-TCN++.

511 They selected as their segmentation model a U-Net [50] network trained from
 512 scratch with the Adam optimizer, cross-entropy loss, learning rate of $1e^{-4}$, and
 513 batch size of 10. Like for the video-based task, workflow recognition was done
 514 by hierarchical ResNet50 followed by a two-stage MS-TCN++.

515 For the video/kinematic-based and video/kinematic/segmentation-based
 516 tasks, the MediCIS team extracted unimodal spatial features using a hierarchical
 517 ResNet50 network for video and segmentation data, followed by concatenation.
 518 Then, they trained a two-stage MS-TCN++.

519 They trained all workflow recognition models with the Adam optimizer,
 520 cross-entropy loss, learning rate of $1e^{-4}$, and batch size of 2. For the hierarchical
 521 ResNet50 network, they emphasized the training for granularities that are harder
 522 to recognize using the following weights in the loss: 1 for phases, 2 for steps, and
 523 5 for both action verbs. They set the number of dilated convolutional layers in
 524 MS-TCN++ to 10, except for the first layer where it was 11. The number of
 525 feature maps for each layer was 64.

526 *3.3. Workflow recognition results*

527 All results were computed on the organizers' hardware via the provided Docker
 528 images. This section only presents the results used for the ranking (balanced
 529 AD-Accuracy). Other results, such as application-dependent scores for each

530 sequence and task, for each participating team, are available as supplementary
 531 material and at www.synapse.org/PETRAW.

532 3.3.1. Task 1: Video-based workflow recognition

533 Task 1 consisted of recognizing phases, steps, and hand verbs using video data
 534 only. Table 3 summarizes the algorithms used by the five teams that submitted
 535 models for this task.

| Team | Hutom * | MedAIR * | NCC Next * | SK | MediCIS |
|---------------|----------------------------------|---------------|--------------------------|---------------|------------------------|
| Preprocessing | X | X | X | X | X |
| Augmentation | X | X | | X | |
| Model | 3DResNet | Trans-SVNet | Xception | ResNet18 | ResNet50 & MS-TCN++ |
| Optimizer | Adam | SGD | Radam | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross-entropy | cross-entropy | cross-entropy | cross-entropy |
| Learning Rate | $1e^{-3}$ | $5e^{-4}$ | $1e^{-3}$ & $1e^{-4}$ | $7.2e^{-4}$ | $1e^{-4}$ |
| Causal | | | | | X |

Table 3: Algorithms used for task 1. Teams that resubmitted models are highlighted with an asterisk. An “X” means that the method performed preprocessing, data augmentation, or is causal.

536 Comparison of the mean AD accuracy values for each test sequence (all
 537 models) (Figure 5) showed only a slight performance decrease (from 95.1% to
 538 82.2%), but sequences 79 and 54 displayed the lowest performance (77.7% and
 539 72.9%, respectively). Moreover, for all the test sequences, one model displayed
 540 lower AD-Accuracy values than the other models.

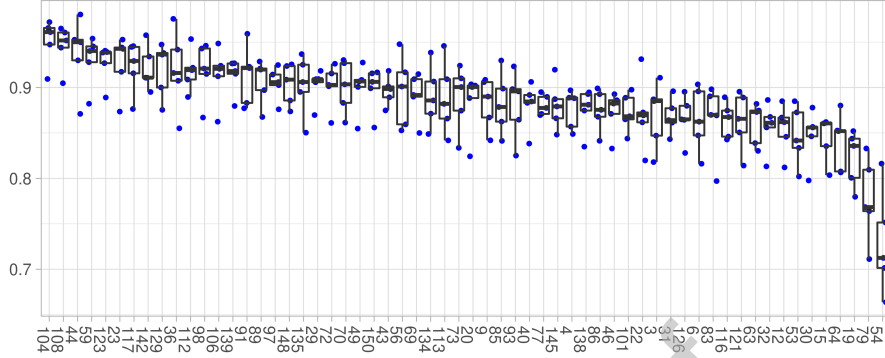


Figure 5: Task 1 recognition AD-Accuracy values (%) for each sequence. Each dot represents the AD-Accuracy of one model. The x-axis represent the test sequence id.

541 Comparison of the mean AD-Accuracy value for each model (Figure 6 showed
 542 that team SK and team Hutom, obtained the highest values ($>90\%$), followed
 543 by team MediCIS and team NCC NEXT ($>87\%$). MedAIR obtained the lowest
 544 results ($\approx 84\%$).

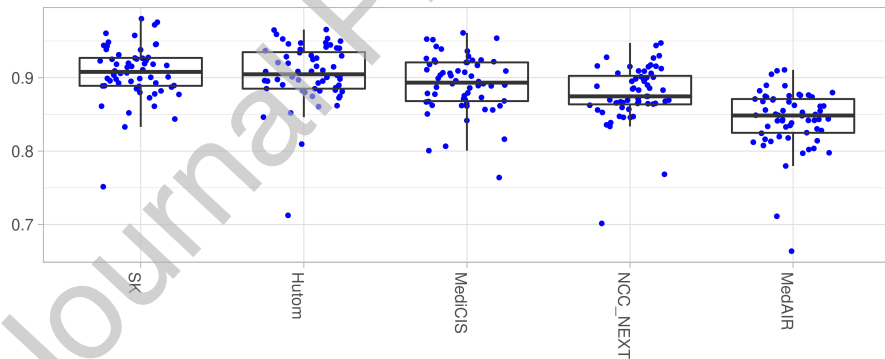


Figure 6: Mean task 1 recognition AD-Accuracy for each model. Each dot represents the AD-Accuracy for one sequence.

545 Team ranking was not influenced by the chosen method (Figure 7), except
 546 for the ranking of the SK and Hutom teams using the rankThenMedian and
 547 testBased methods.

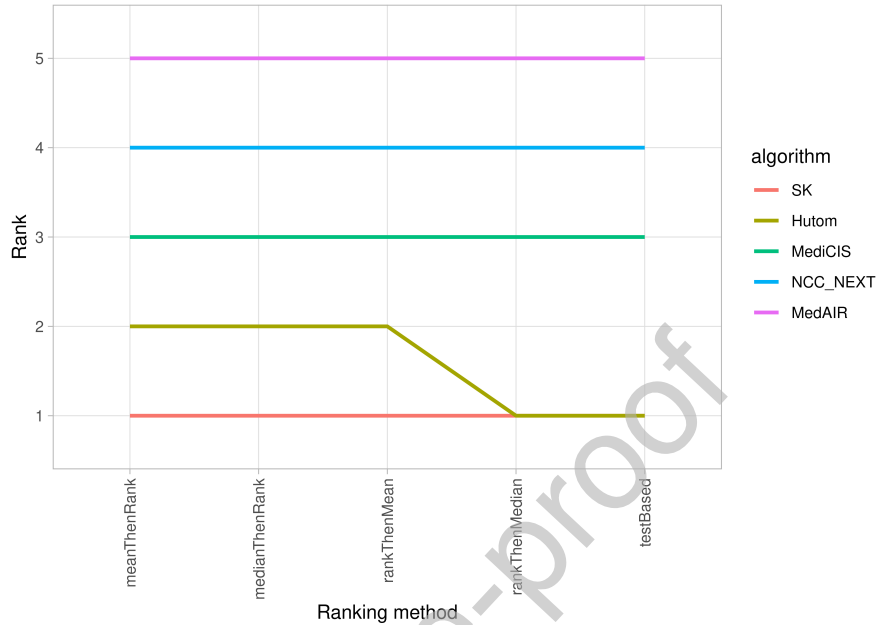


Figure 7: Task 1 recognition ranking stability using different ranking methods. Rank 1 indicates the best method.

548 3.3.2. Task 2: Kinematic-based workflow recognition

549 Task 2 consisted of recognizing phases, steps, and hand verbs using kinematic
 550 data only. Table 4 summarizes the methods used by the six participating teams
 551 for this task.

552 As with task 1, the performance per sequence slightly decreased (Figure 8).
 553 The highest AD-Accuracy values were superior to 90% for all teams. Three
 554 sequences (including sequences 79 and 54) had mean AD-Accuracy values inferior
 555 to 80%. Unlike task 1, the majority of sequences did not have outliers.

| Team | Hutom | JHU-CIRL | MedAIR | NCC Next | SK | MediCIS |
|---------------|-------------------------------|---------------|---------------|-----------------------|---------------|---------------|
| Preprocessing | X | X | X | X | X | X |
| Augmentation | | | | | | |
| Model | Bi-LSTM | Uni-LSTM | Trans-SVNet | LightGBM | Stacked-LSTM | MS-TCN++ |
| Optimizer | Adam | Adam | SGD | Gradient Boosting | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross-entropy | cross-entropy | MAE | cross-entropy | cross-entropy |
| Learning Rate | $1e^{-3}$ | $1e^{-3}$ | $5e^{-4}$ | $1e^{-1}$ & $5e^{-2}$ | $1.5e^{-3}$ | $1e^{-4}$ |
| Causal | | X | | X | X | |

Table 4: Summary of the models used for task 2. An “X” means that the method performed preprocessing, data augmentation, or is causal.

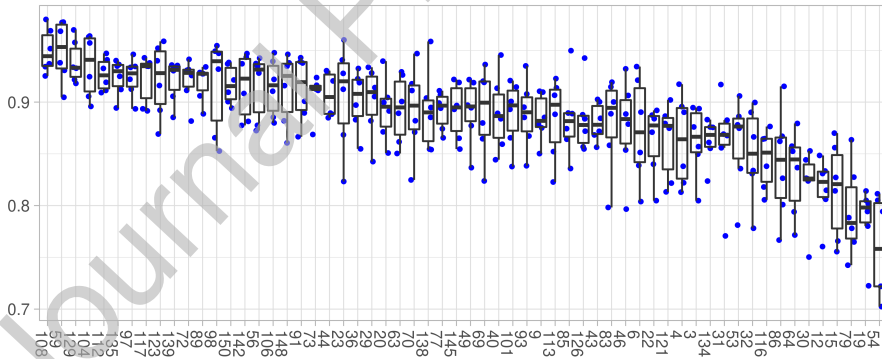


Figure 8: Task 2 recognition AD-Accuracy for each sequence. Each dot represents the AD-Accuracy of one model.

556 Results were very similar among teams (Figure 9). Four had a mean AD-
557 Accuracy value of between 89.7% and 90.7%, and the other two displayed mean
558 AD-accuracy values of 86.4% and 84.3%, respectively.

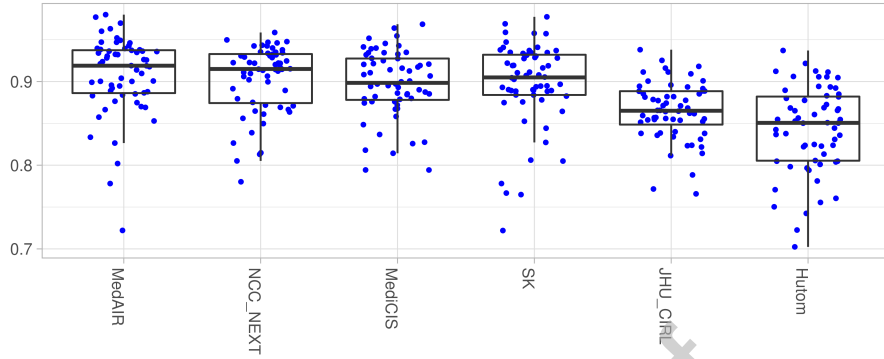


Figure 9: Mean task 2 recognition AD-Accuracy for each model. Each dot represents the AD-Accuracy for one sequence.

559 Ranking was not stable for team SK and team MediCIS (Figure 10). As
 560 MediCIS was a non-competing team, SK was ranked third for this task.

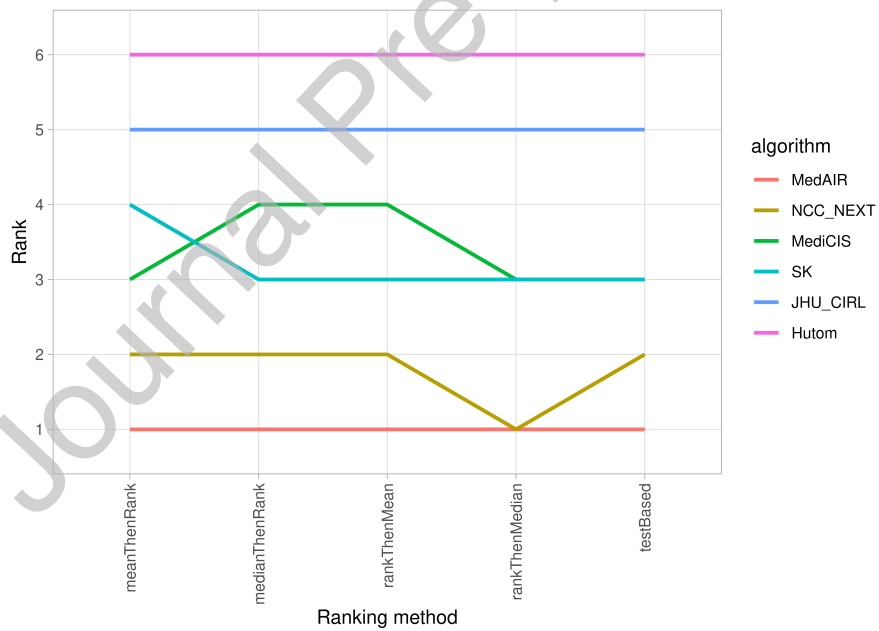


Figure 10: Task 2 recognition ranking stability using the indicated ranking methods.

561 *3.3.3. Task 3: Segmentation-based workflow recognition*

562 Task 3 consisted of recognizing phases, steps, and hand verbs using semantic
563 segmentation data only. First, the results of the segmentation models provided
564 by the participants will be described, and then the workflow recognition models.

565 *Segmentation models:*

566 Table 5 summarizes the methods used by the four participating teams to
perform semantic segmentation.

| Team | Hutom * | NCC Next * | SK | MediCIS |
|---------------|----------------------------------|---------------|---------------|---------------|
| Preprocessing | X | X | X | X |
| Augmentation | X | | X | |
| Model | DeepLabV3+ | DeepLabV3+ | U-Net | U-Net |
| Optimizer | Adam | Radam | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross-entropy | cross-entropy | cross-entropy |
| Learning Rate | $1e^{-3}$ | $1e^{-4}$ | $2.4e^{-5}$ | $1e^{-4}$ |

Table 5: Segmentation models used for task 3. Teams that resubmitted models are highlighted with an asterisk. An “X” means that the method performed preprocessing, data augmentation, or is causal.

567

568 Comparison of the IoU values for each class independently and for all classes
569 (Macro) (Table 6) showed that, the IoU varied between 94.0% and 91.1% for
570 Macro. Pegs were the least recognized structure (IoU between 83.9% and 82.3%).
571 Specific sequences with lower performance were not identified.

572 Comparison of the mean IoU values of each team for all classes (Macro) and
573 for each class independently (Table 7) showed similar Macro results for the NCC
574 Next, SK and MediCIS teams (96.9%, 96.4%, and 94.0%, respectively). The
575 Hutom team’s Macro IoU was the lowest (85.0%), mainly due to the IoU for
576 pegs (63.3%). Figure 11 presents the ground truth and the segmentation results
577 of each team for one frame.

578 *Workflow models*

| | Mean | Median | Max | Min |
|------------|------|--------|------|------|
| Background | 98.8 | 98.9 | 98.9 | 98.7 |
| Base | 96.1 | 96.2 | 96.3 | 95.6 |
| Pegs | 83.2 | 83.1 | 83.9 | 82.3 |
| Blocks | 91.7 | 91.7 | 92.5 | 90.8 |
| Left tool | 94.9 | 95.3 | 97.6 | 87.3 |
| Right tool | 94.0 | 94.5 | 96.9 | 88.9 |
| Macro | 93.1 | 93.2 | 94.0 | 91.1 |

Table 6: Mean Intersection-Over-Union values for all classes of each sequence independently

| | Hutom * | NCC Next * | SK | MediCIS |
|------------|---------|-------------|-------------|---------|
| Background | 97.7 | 99.5 | 99.2 | 98.9 |
| Base | 91.4 | 98.4 | 98.4 | 96.1 |
| Pegs | 63.3 | 92.1 | 92.0 | 85.3 |
| Blocks | 82.8 | 96.0 | 96.0 | 92.2 |
| Left tool | 89.3 | 98.1 | 96.1 | 96.0 |
| Right tool | 85.5 | 97.8 | 96.7 | 95.8 |
| Macro | 85.0 | 96.9 | 96.4 | 94.0 |

Table 7: Mean Intersection-Over-Union values for all the classes of each team. Teams that resubmitted models are highlighted with an asterisk and best results are in bold.

579 Table 8 summarizes the methods used by the four participating teams to
 580 perform the workflow recognition.

581 Comparison of the mean AD-Accuracy values for each test sequence (Figure
 582 12) showed that performance decreased from 87.5% to 76.6%. The same two
 583 sequences (79 and 54) displayed very low results (67.4% and 65.5%, respectively).
 584 Moreover, for all test cases, one model had results lower than 70%.

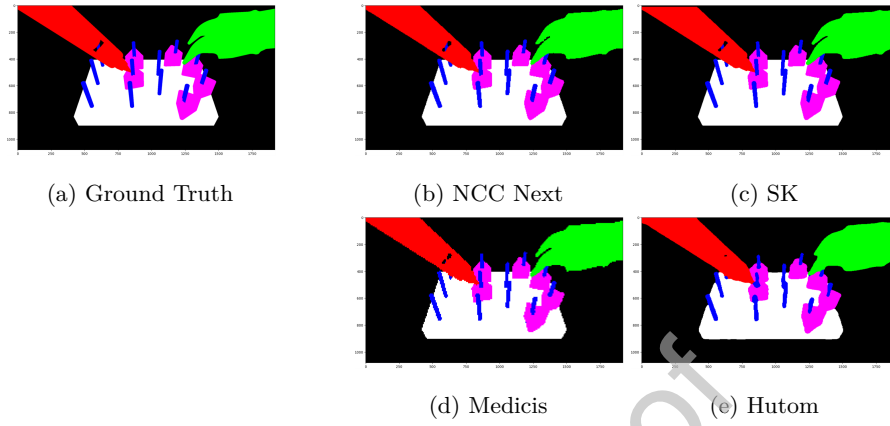


Figure 11: Ground truth (a) and segmentation results for each team (b to e) for one frame.

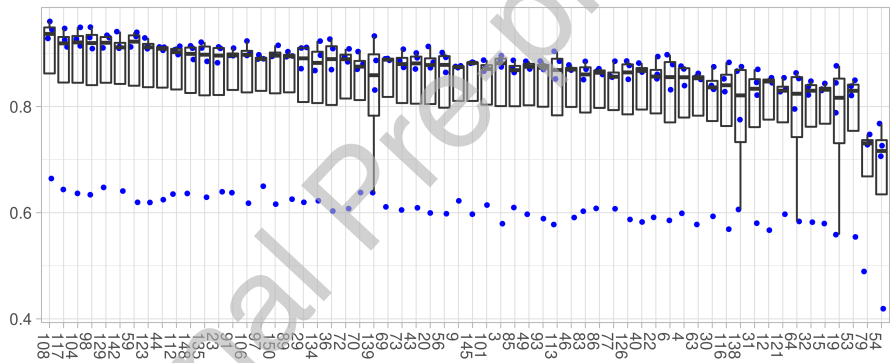


Figure 12: Task 3 recognition AD-Accuracy for each sequence. Each dot represents the AD-Accuracy of one model.

585 Comparison of the mean AD-Accuracy value for each model indicated that
 586 three teams obtained results between 88.5% and 87.2%, whereas the Hutom
 587 team had a mean AD-Accuracy value of 60.3% (Figure 13).

| Team | Hutom * | NCC Next * | SK | MediCIS |
|---------------|----------------------------------|----------------|---------------|------------------------|
| Preprocessing | X | X | X | X |
| Augmentation | X | | X | |
| Model W | SlowFast50 | EfficientNetB7 | ResNet18 | ResNet50 & MS-TCN++ |
| Optimizer | Adam | Radam | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross-entropy | cross-entropy | cross-entropy |
| Learning Rate | $1e^{-3}$ | $1e^{-4}$ | $1e^{-4}$ | $1e^{-4}$ |
| Causal | | | | X |

Table 8: Summary of the models used for task 3 (segmentation-based workflow recognition). Teams that resubmitted models are highlighted with an asterisk. An “X” means that the method performed preprocessing, data augmentation, or is causal.

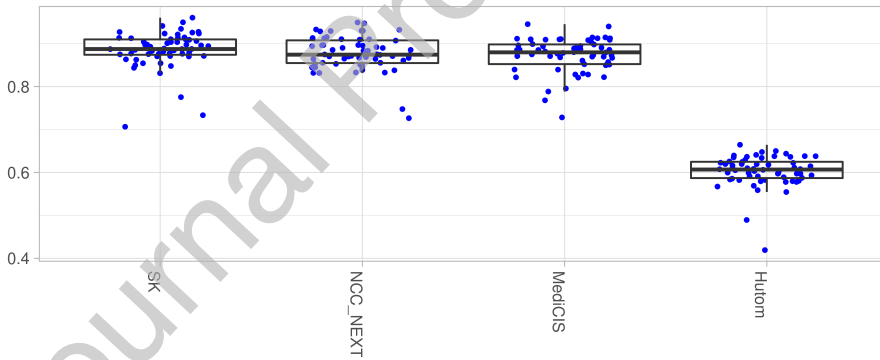


Figure 13: Mean recognition AD-Accuracy for each model for task 3. Each dot represents the AD-Accuracy for one sequence.

588 The choice of method did not influence the team ranking, except for the
589 second (NCC NEXT) and the third (MediCIS) rank (Figure 14).

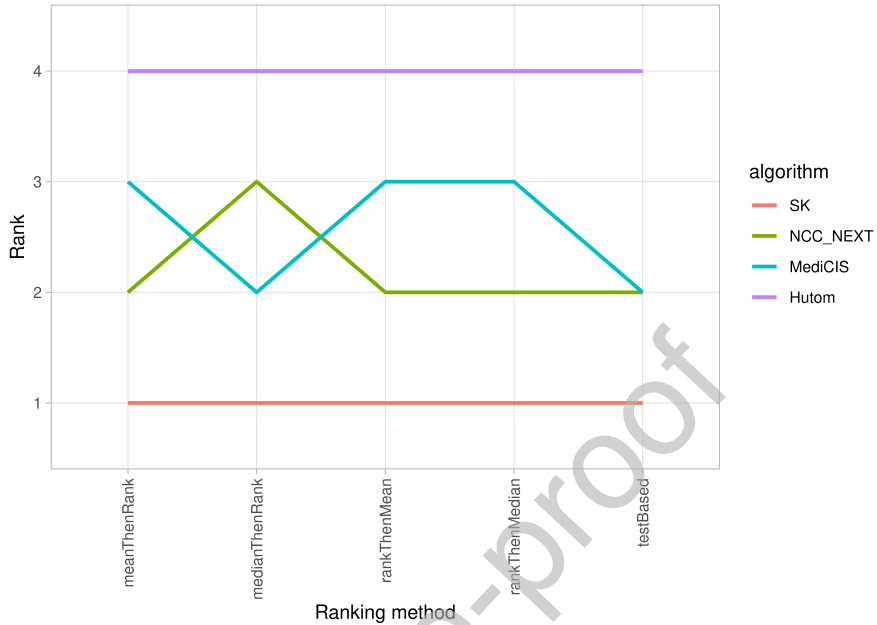


Figure 14: Task 3 recognition ranking stability using the indicated ranking methods.

590 3.3.4. Task 4: Video/kinematic-based workflow recognition

591 Task 4 consisted of recognizing phases, steps, and hand verbs using video and
 592 kinematic data. Table 9 summarizes the methods used by the six participating
 593 teams.

594 AD-Accuracy values for each sequence were similar to those of the previous
 595 tasks (Figure 15). Indeed, performance slightly decreased from 95.1% to 83.1%
 596 for most sequences, and was again low for sequences 79 and 54 (81.2% and
 597 76.5%). For this task, the number of outliers was limited.

| Team | Hutom * | MedAIR | MMLAB | NCC NEXT * | SK | MediCIS |
|---------------|-------------------------------------|-------------------|--------------------------|--|--------------------------------|------------------------|
| Preprocessing | X | X | X | X | X | X |
| Augmentation | X | X | | | X | |
| Model | 3D ResNet & Bi-LSTM | MRG-Net & CNN | ResNet50 & LSTM | Xception & LightGBM | ResNet18 & Stacked -LSTM | ResNet50 & MS-TCN++ |
| Optimizer | Adam | Adam | Adam | Radam & Gradient Boosting | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross- entropy | cross- entropy | cross- entropy & MAE | cross- entropy | cross- entropy |
| Learning Rate | $1e^{-3}$ | $1e^{-4}$ | $1e^{-3}$ & $1e^{-2}$ | $1e^{-1}$ & $5e^{-2}$ & $1e^{-3}$ & $1e^{-4}$ | $7.2e^{-4}$ & $1.5e^{-3}$ | $1e^{-4}$ |
| Causal | | | X | | | |

Table 9: Summary of the models used for task 4. Teams that resubmitted models are highlighted with an asterisk. An “X” means that the method performed preprocessing, data augmentation, or is causal.

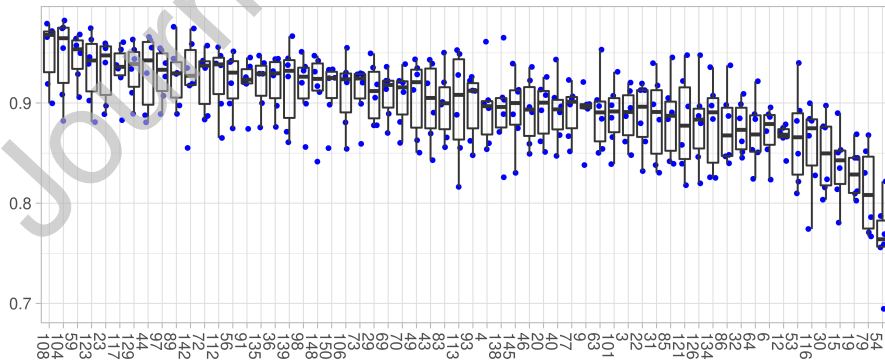


Figure 15: Task 4 recognition AD-Accuracy values for each sequence. Each dot represents the AD-Accuracy for one model.

598 The NCC NEXT team obtained the best results (Figure 16), with a mean
 599 AD-Accuracy of 93.1%, followed by SK, Hutom, and MediCIS teams with results
 600 of between 91.6% and 90.2%. For the last two teams, the AD-Accuracy was
 601 above 84.5%.

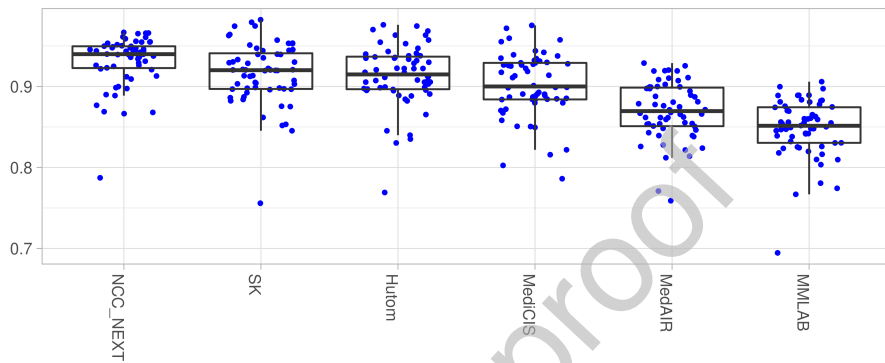


Figure 16: Mean task 4 recognition AD-Accuracy for each team. Each dot represents the AD-Accuracy for one sequence.

602 The ranking is stable according to the ranking method chosen (Figure 17).

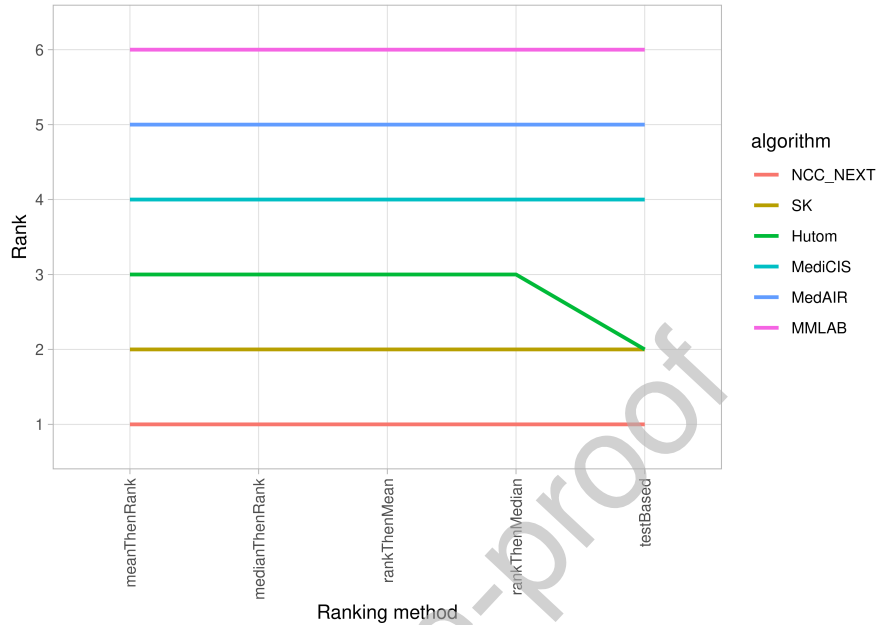


Figure 17: Task 4 recognition ranking stability using the indicated ranking methods.

603 3.3.5. Task 5: Video/kinematic/segmentation-based workflow recognition

604 In task 5, teams recognized phases, steps, and hand verbs using video,
 605 kinematic and segmentation data. Table 10 summarizes the recognition methods
 606 used by the four participating teams. The models to create the segmentation
 607 were the same as those described in Table 5.

608 As for the previous tasks, the mean AD-Accuracy values per sequence (Figure
 609 18) highlighted a slight performance decrease (from 97.2% to 85.9%). Sequences
 610 79 and 54 again displayed the lowest performances (80.8% and 78.0%, respec-
 611 tively).

| Team | Hutom * | NCC NEXT * | SK | MediCIS |
|---------------|----------------------------------|--|--|------------------------|
| Preprocessing | X | X | X | X |
| Augmentation | X | | X | |
| Model | 3D ResNet & Bi-LSTM | Xception, EfficientNetB7 & LightGBM | ResNet18 & Staked-LSTM | ResNet50 & MS-TCN++ |
| Optimizer | Adam | Radam & Gradient Boosting | Adam | Adam |
| Loss | Equalization v2 & Normsoftmax | cross-entropy & MAE | cross-entropy | cross-entropy |
| Learning Rate | $1e^{-3}$ | $1e^{-1}$, $5e^{-2}$, $1e^{-3}$ & $1e^{-4}$ | $7.2e^{-4}$, $1.5e^{-3}$ & $1e^{-4}$ | $1e^{-4}$ |
| Causal | | | | |

Table 10: Models used for task 5. Teams that resubmitted models are highlighted with an asterisk. An “X” means that the method performed preprocessing, data augmentation, or is causal.

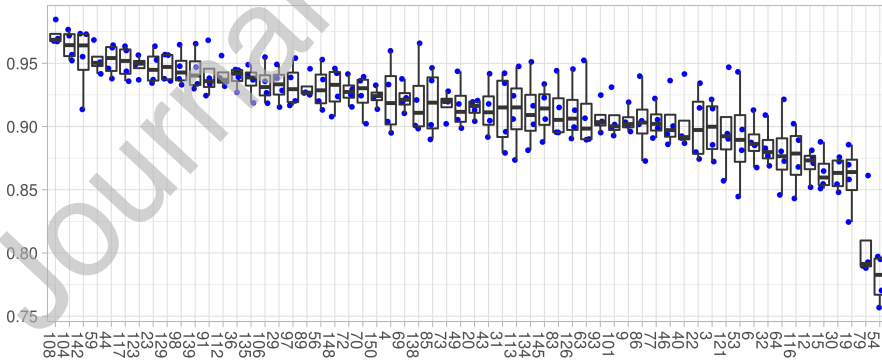


Figure 18: Task 5 AD-Accuracy for each sequence. Each dot represents the AD-Accuracy for one model.

612 The teams’ mean AD-Accuracy values ranged between 93.1% and 89.8%
613 (Figure 19). The SK and Hutom teams displayed very similar results, with 91.4%

614 and 91.3%, respectively. However, the chosen ranking method did not influence
 615 the final rank (Figure 20).

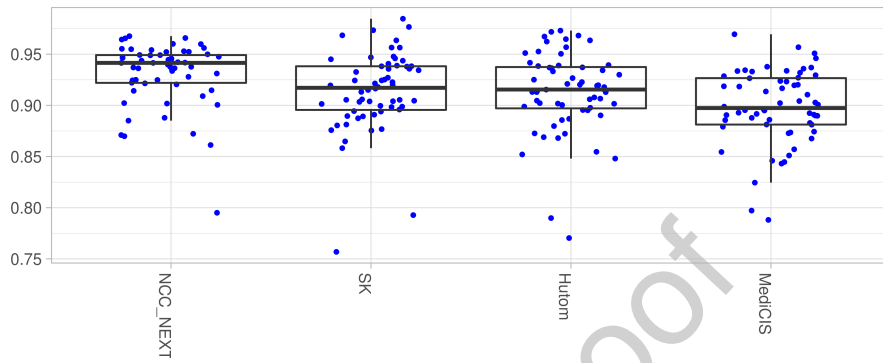


Figure 19: Average task 5 recognition AD-Accuracy for each team. Each dot represents the AD-Accuracy for one sequence.

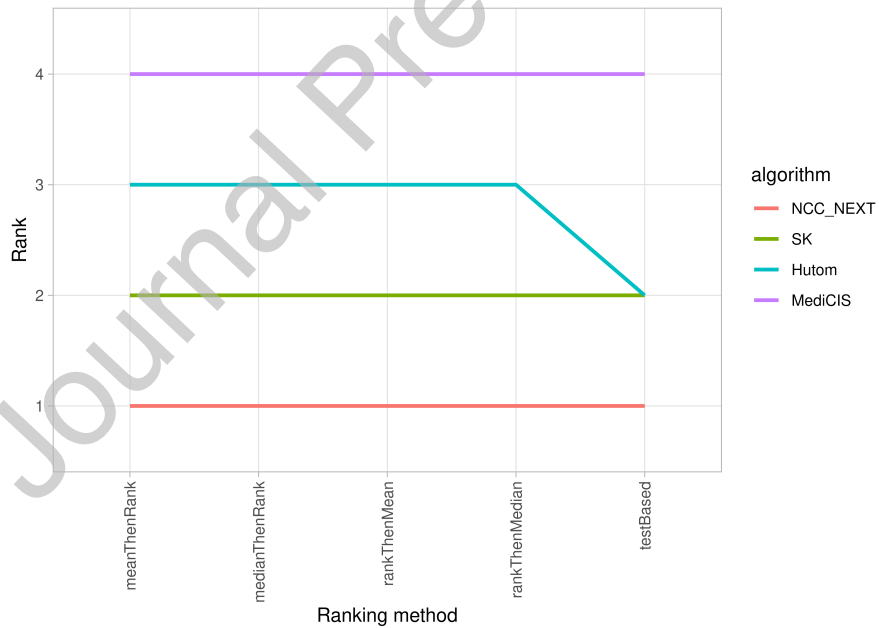


Figure 20: Task 5 ranking stability using the indicated ranking methods.

616 *3.3.6. Workflow recognition results summary*

617 Table 11 summarizes the results of each team for the five tasks. All the best
618 methods displayed mean AD-Accuracy superior to 90%, except for task 3.

| Team | Task 1 | Task 2 | Task 3 | Task 4 | Task 5 |
|----------|--------------|--------------|--------------|----------------|----------------|
| Hutom | 90.51 * | 84.31 | 60.28 * | 91.33 * | 91.27 * |
| JHU-CIRL | | 86.45 | | | |
| MedAIR | 84.31 * | 90.72 | | 86.98 | |
| MMLAB | | | | 84.80 | |
| NCC NEXT | 87.77 * | 90.32 | 87.71 * | 93.09 * | 93.09 * |
| SK | 90.77 | 89.66 | 88.51 | 91.61 | 91.37 |
| MediCIS | 89.15 | 89.71 | 87.22 | 90.18 | 89.81 |

Table 11: Mean AD-Accuracy of each team for the five tasks. The best results are highlighted in bold for each task. Resubmitted models are highlighted with an asterisk.

619 *3.4. Additional analyses*

620 The additional analyses concern four of the seven participating teams: Hutom,
621 NCC Next, SK, and MediCIS. They were the only teams to participate with a
622 combination of the same or similar models used for the unimodal tasks. Although
623 MedAIR team participated in task 4 and the two corresponding unimodal tasks
624 (1 and 2), the models used were too different to allow a model comparison.

625 *3.4.1. Comparison between unimodal and multimodal models*

626 Table 12 presents the results of the statistical analysis. For the four teams,
627 the combination of video and kinematics (task 4) is statistically different than
628 the use of only one modality (tasks 1 and 2). The same statistical differences
629 exist between the combination of the three modalities (task 5) and each modality
630 individually (tasks 1, 2, and 3), with the exception of task 2 and task 5 for the
631 MediCIS team. However, the addition of the segmentation modality (task 5) to
632 the video/kinematic-based (task 4) models was only significant for the MediCIS
633 team.

| Team | Hutom | NCC NEXT | SK | MediCIS |
|----------|-------|----------|----|---------|
| T1 <> T4 | X | X | X | X |
| T2 <> T4 | X | X | X | X |
| T1 <> T5 | X | X | X | X |
| T2 <> T5 | X | X | X | |
| T3 <> T5 | X | X | X | X |
| T4 <> T5 | | | | X |

Table 12: Significant performance differences between unimodal and multimodal tasks. T1 <> T4: comparison of task 1 and task 4; X: significant performance variation (p-value < 0.05).

634 3.4.2. Execution time

635 Table 13 presents the execution time for the four teams and each task. For
636 NCC Next team, the duration could not be determined because the predictions
637 were locally written at the end of the Docker image execution. Execution time
638 was highly variable among the teams, with the shortest (except task 2) achieved
639 by the SK team. The shortest execution times overall were obtained for task 2
640 (3 min for SK and less than 1 minute for the Hutom and MediCIS teams).

| Team | Hutom | NCC NEXT | SK | MediCIS |
|--------|------------|----------|---------|---------|
| Task 1 | 56 min | CBD | 50 min | 202 min |
| Task 2 | < 1 min | CBD | 3 min | < 1 min |
| Task 3 | 13 550 min | CBD | 145 min | 725 min |
| Task 4 | 57 min | CBD | 53 min | 203 min |
| Task 5 | 13 600 min | CBD | 175 min | 928 min |

Table 13: Execution times to compute the results of the 60 test sequences. CBD: Could not Be Determined

641 4. Discussion

642 Accurate surgical workflow recognition is necessary for context-aware computer-
643 assisted surgical systems. The proposed methods obtained good results but
644 were not perfect and the PETRAW data set itself presented several limitations.
645 Specifically, the peg transfer task is significantly easier than a real surgical
646 intervention due to the simpler environment, clearly identifiable objects, static
647 field of view, and constant lighting. In addition, each sequence was performed
648 by the same operator resulting in lower data set variability.

649 By analyzing the performance of the methods across individual sequences,
650 we observed a gradual decrease in performance, except for two sequences (54
651 and 79) that displayed very low AD-Accuracy compared to the others regardless
652 of modality. We analyzed these two sequences in detail to understand this poor
653 performance. In sequence 54, the block was dropped twice during the transfer
654 between hands, forcing the operator to catch the block for a second time. In
655 addition, one block got stuck on the peg, forcing the operator to reposition it.
656 Sequence 79 is one of the sequences identified as containing uncertainty (see
657 Section 2.3.3). However, the overlapping steps (by 0.5 seconds) could not entirely
658 explain the low performance, as the overlap was partially compensated by the
659 delay of 0.25 seconds used to compute the AD-Accuracy. In addition, a block got
660 stuck on a peg in this sequence and the order in which the blocks were caught
661 did not correspond to the one used in most sequences. These deviations from
662 the most common workflow might explain the low performance.

663 For task 1 (video-based recognition), ResNet-based models gave the best
664 results, and the simplest model was ranked first. For task 2 (kinematic-based
665 recognition), LSTM-based methods presented the worst results. For task 3,
666 the two segmentation models used (DeepLabV3 and U-Net), displayed similar
667 IoU values and the differences were probably due to differences in the training
668 characteristics. For workflow recognition, the EfficientNetB7 and ResNet models
669 obtained similar results. For Tasks 4 and 5, the NCC NEXT team's strategy
670 (i.e., using the modality that gave the best results in the unimodal tasks for each

671 workflow component) provided the best result.

672 For the segmentation-based recognition task (task 3), the segmentation quality
673 seemed to influence workflow recognition up to a certain threshold. Indeed, the
674 workflow recognition performances of the three teams with Macro IoU values
675 superior to 94.0% were similar (AD-Accuracy between 88.5% and 87.2%), but
676 the ranking was inverted for the two first teams. Conversely, the workflow
677 recognition performance with a Macro IoU value of 85% dropped drastically
678 (60.3%). Additional research is required to fully quantify and understand the
679 degree to which segmentation quality influences workflow recognition since, in
680 this challenge, teams used different combinations of models for the segmentation
681 and workflow recognition components.

682 For tasks 1 to 4, at least one team submitted a method that could be truly
683 causal. It is important to note that several proposed methods were provably
684 non-causal due to their preprocessing steps and not the core network such as
685 with NCC NEXT (task 3), SK (task 1, 3, 4, 5), and MediCIS (task 2, 4 and
686 5). Causal methods generally have lower performance than non-causal models.
687 With the exception of task 4, the causal methods displayed performances that
688 were surprisingly close to that of the best method. For example, for task 2, the
689 AD-Accuracy of the best method was 90.7%, compared to 90.3% and 89.7%
690 for the causal methods by NCC NEXT and SK, respectively. Obviously, it is
691 not possible to conclude that causal methods give similar results to acausal
692 models: i) because during the challenge we did not have the two versions of a
693 similar method, ii) due to data simplicity. Nevertheless, the results of the causal
694 methods are promising for developing applications, such as the implementation
695 of automatic reports after training sessions on a virtual simulator.

696 Among the seven participating teams, four (Hutom, NCC Next, SK, and
697 MediCIS) participated in the multimodal tasks (4 and 5) with a combination of
698 the same or similar models used for the unimodal tasks. In all cases, recognition
699 was improved when several modalities were used (Table 11); however, the
700 addition of segmentation modality decreased the performance. The statistical
701 analysis (Table 12) confirmed a significant performance improvement when using

702 multimodal models, with the exception of tasks 2 and 5 for the MediCIS team.
703 The performance decrease experienced with the addition of the segmentation
704 modality to the video/kinematic-based models was only significant for the
705 MediCIS team.

706 Therefore, the combination of video and kinematic (task 4) data gives sig-
707 nificantly better results compared with other modality combinations. The
708 results obtained by the MedAIR team could contradict this point because they
709 obtained better results for the kinematic-based recognition task than for the
710 video/kinematic-based one. However, the models they used were very different:
711 a Trans-SVNet and an MRG-Net combined with a CNN respectively. So, in this
712 case, it is difficult to determine if the performance modifications were due to the
713 model or to the modalities used. However, task 4 was more time-consuming than
714 task 2 (53 vs. 3 minutes for SK, 57 vs. less than 1 for Hutom, and 203 vs. less
715 than 1 for MediCIS). One may ask whether it is wise to spend 2,000% to 20,000%
716 more computing time for less than a 3% improvement. The training time should
717 also be taken into account, as it is much more time-consuming [51, 52], but we
718 did not have access to this information. Data storage should also be considered.
719 Video can require a lot of storage space, especially for long surgical interventions.
720 Conversely, kinematic data are less voluminous.

721 Future work should focus on overcoming the limitations of the current data
722 set by including peg transfer sequences performed by several operators in different
723 systems. Moreover, tests on more realistic data are necessary to validate the
724 finding that kinematic data display the best performances in recognition rate
725 and have less environmental impact thanks to the lowest computation time and
726 storage cost.

727 **Acknowledgements**

728 Authors thanks IRT b<>com for providing the “Surgery Workflow Toolbox
729 [annotate]” software, used for this work.

730 **Statements of ethical approval**

731 All procedures performed in studies involving human participants were in
732 accordance with the ethical standards of the institutional and/or national research
733 committee and with the 1964 Helsinki declaration and its later amendments or
734 comparable ethical standards. This article does not contain patient data.

735 **Conflict of interest statement**

736 The authors declare that they have no conflict of interest.

737 **References**

- 738 [1] P. Jannin, M. Raimbault, X. Morandi, and B. Gibaud. Modeling surgical
739 procedures for multimodal image-guided neurosurgery. In *Lecture Notes in*
740 *Computer Science (including subseries Lecture Notes in Artificial Intelligence*
741 *and Lecture Notes in Bioinformatics)*, volume 2208, pages 565–572. Springer
742 Verlag, 10 2001.
- 743 [2] Florent Lalys and Pierre Jannin. Surgical process modelling: a review.
744 *International Journal of Computer Assisted Radiology and Surgery*, 9(3):495–
745 511, 11 2013.
- 746 [3] F Despinoy, D Bouget, G Forestier, C Penet, N Zemiti, P Poignet, and
747 P Jannin. Unsupervised trajectory segmentation for surgical gesture recog-
748 nition in robotic training. *IEEE Transactions on Biomedical Engineering*,
749 63(6):1280–1291, 8 2015.
- 750 [4] Arnaud Huaultmé, Kanako Harada, Germain Forestier, Mamoru Mitsui-
751 shi, and Pierre Jannin. Sequential surgical signatures in micro-suturing
752 task. *International Journal of Computer Assisted Radiology and Surgery*,
753 13(9):1419–1428, 5 2018.
- 754 [5] Germain Forestier, Laurent Riffaud, François Petitjean, Pierre Louis Henaux,
755 and Pierre Jannin. Surgical skills: Can learning curves be computed from

- 756 recordings of surgical activities? *International Journal of Computer Assisted*
757 *Radiology and Surgery*, 13(5):629–636, 5 2018.
- 758 [6] S.-Y. Ko, J Kim, W.-J. Lee, and D.-S. Kwon. Surgery task model for
759 intelligent interaction between surgeon and laparoscopic assistant robot.
760 *International Journal of Assitive Robotics and Mechatronics*, 8(1):38–46, 10
761 2007.
- 762 [7] Warren S. Sandberg, Bethany Daily, Marie Egan, James E. Stahl, Julian M.
763 Goldman, Richard A. Wiklund, and David Rattner. Deliberate Perioperative
764 Systems Design Improves Operating Room Throughput:. *Anesthesiology*,
765 103(2):406–418, 10 2005.
- 766 [8] Beenish Bhatia, Tim Oates, Yan Xiao, and Peter Hu. Real-time identifi-
767 cation of operating room state from video. In *Proceedings of the National*
768 *Conference on Artificial Intelligence*, volume 2, pages 1761–1766, 10 2007.
- 769 [9] Gwenole Quellec, Mathieu Lamard, Beatrice Cochener, and Guy Cazuguel.
770 Real-Time Task Recognition in Cataract Surgery Videos Using Adaptive Spa-
771 tiotemporal Polynomials. *IEEE Transactions on Medical Imaging*, 34(4):877–
772 887, 4 2015.
- 773 [10] Arnaud Huauhmé, Pierre Jannin, Fabian Reche, Jean-Luc Faucheron, Alexan-
774 dre Moreau-Gaudry, and Sandrine Voros. Offline identification of surgical
775 deviations in laparoscopic rectopexy. *Artificial Intelligence in Medicine*,
776 104:1–26, 9 2019.
- 777 [11] A Huauhmé, F Despinoy, S A Heredia Perez, K Harada, M Mitsuishi, and
778 P Jannin. Automatic annotation of surgical activities using virtual reality
779 environments. *International Journal of Computer Assisted Radiology and*
780 *Surgery*, 14(10):1663–1671, 7 2019.
- 781 [12] Nicolas Padoy, Tobias Blum, S.-A. Seyed Ahmad S.-A. Seyed Ahmad
782 Ahmadi, Hubertus Feussner, Marie Odile M.-O. Marie Odile M.-O. Berger,

- 783 and Nassir Navab. Statistical modeling and recognition of surgical workflow.
784 *Medical Image Analysis*, 16(3):632–641, 1 2010.
- 785 [13] Andru P. Twinanda, Sherif Shehata, Didier Mutter, Jacques Marescaux,
786 Michel De Mathelin, and Nicolas Padoy. EndoNet: A Deep Architecture for
787 Recognition Tasks on Laparoscopic Videos. *IEEE Transactions on Medical*
788 *Imaging*, 36(1):86–97, 2 2016.
- 789 [14] L Bouarfa, P P Jonker, and J Dankelman. Discovery of high-level tasks in
790 the operating room. *Journal of Biomedical Informatics*, 44(3):455–462, 10
791 2011.
- 792 [15] A James, D Vieira, B Lo, A Darzi, and G.-Z. Yang. Eye-Gaze Driven Surgical
793 Workflow Segmentation. *Medical Image Computing and Computer-Assisted*
794 *Intervention MICCAI 2007*, pages 110–117, 11 2007.
- 795 [16] Florent Lalys, David Bouget, Laurent Riffaud, and Pierre Jannin. Auto-
796 matic knowledge-based recognition of low-level tasks in ophthalmological
797 procedures. *International Journal of Computer Assisted Radiology and*
798 *Surgery*, 8(1):39–49, 4 2012.
- 799 [17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet clas-
800 sification with deep convolutional neural networks. *Advances in neural*
801 *information processing systems*, 25, 2012.
- 802 [18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual
803 learning for image recognition. In *Proceedings of the IEEE Computer*
804 *Society Conference on Computer Vision and Pattern Recognition*, volume
805 2016-Decem, pages 770–778, 2016.
- 806 [19] Sepp Hochreiter and Jürgen Schmidhuber. Long Short-Term Memory.
807 *Neural Computation*, 9(8):1735–1780, 11 1997.
- 808 [20] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua
809 Bengio. On the properties of neural machine translation: Encoder-decoder
810 approaches. *arXiv preprint arXiv:1409.1259*, 2014.

- 811 [21] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones,
812 Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you
813 need. *Advances in neural information processing systems*, 30, 2017.
- 814 [22] Duygu Sarikaya and Pierre Jannin. Surgical Gesture Recognition with
815 Optical Flow only. *arXiv*, 4 2019.
- 816 [23] Isabel Funke, Sebastian Bodenstedt, Florian Oehme, Felix von Bechtol-
817 sheim, Jürgen Weitz, and Stefanie Speidel. Using 3D Convolutional Neural
818 Networks to Learn Spatiotemporal Features for Automatic Surgical Gesture
819 Recognition in Video. In *Lecture Notes in Computer Science (including*
820 *subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioin-*
821 *formatics)*, volume 11768 LNCS, pages 467–475, 2019.
- 822 [24] Robert DiPietro and Gregory D. Hager. Automated Surgical Activity
823 Recognition with One Labeled Sequence, 10 2019.
- 824 [25] Arnaud Huauilmé, Duygu Sarikaya, Kévin Le Mut, Fabien Despinoy, Yong-
825 hao Long, Qi Dou, Chin-Boon Chng, Wenjun Lin, Satoshi Kondo, Laura
826 Bravo-Sánchez, Pablo Arbeláez, Wolfgang Reiter, Manoru Mitsuishi, Kanako
827 Harada, and Pierre Jannin. Micro-surgical anastomose workflow recogni-
828 tion challenge report. *Computer Methods and Programs in Biomedicine*,
829 212:106452, 11 2021.
- 830 [26] Yong-Hao Long, Jie-Ying Wu, Bo Lu, Yue-Ming Jin, Mathias Unberath,
831 Yun-Hui Liu, Pheng-Ann Heng, and Qi Dou. Relational Graph Learning on
832 Visual and Kinematics Embeddings for Accurate Gesture Recognition in
833 Robotic Surgery. *arXiv*, 2020.
- 834 [27] Yidan Qin, Max Allan, Yisong Yue, Joel W. Burdick, and Mahdi Az-
835 izian. Learning Invariant Representation of Tasks for Robust Surgical State
836 Estimation. *arXiv*, 2 2021.
- 837 [28] S.A Heredia Perez, Kanako Harada, and Mamoru Mitsuishi. Haptic As-

- 838 sistance for Robotic Surgical Simulation. *27th Annual Congress of Japan*
839 *Society of Computer Aided Surgery*, 20(4):232–233, 11 2018.
- 840 [29] O Dergachyova, D Bouget, A Huauilmé, X Morandi, and P Jannin. Auto-
841 matic data-driven real-time segmentation and recognition of surgical work-
842 flow. *International Journal of Computer Assisted Radiology and Surgery*,
843 10 2016.
- 844 [30] Lena Maier-Hein, Matthias Eisenmann, Annika Reinke, Sinan Onogur,
845 Marko Stankovic, Patrick Scholz, Tal Arbel, Hrvoje Bogunovic, Andrew P
846 Bradley, Aaron Carass, Carolin Feldmann, Alejandro F Frangi, Peter M
847 Full, Bram van Ginneken, Allan Hanbury, Katrin Honauer, Michal Kozubek,
848 Bennett A Landman, Keno März, Oskar Maier, Klaus Maier-Hein, Bjoern H
849 Menze, Henning Müller, Peter F Neher, Wiro Niessen, Nasir Rajpoot,
850 Gregory C Sharp, Korsuk Sirinukunwattana, Stefanie Speidel, Christian
851 Stock, Danail Stoyanov, Abdel Aziz Taha, Fons van der Sommen, Ching-
852 Wei Wang, Marc-André Weber, Guoyan Zheng, Pierre Jannin, and Annette
853 Kopp-Schneider. Why rankings of biomedical image analysis competitions
854 should be interpreted with care. *Nature Communications*, 9(1):5217, 12
855 2018.
- 856 [31] Manuel Wiesenfarth, Annika Reinke, Bennett A. Landman, Matthias Eisen-
857 mann, Laura Aguilera Saiz, M. Jorge Cardoso, Lena Maier-Hein, and
858 Annette Kopp-Schneider. Methods and open-source toolkit for analyzing
859 and visualizing challenge results. *Scientific Reports*, 11(1):2369, 12 2021.
- 860 [32] Pierre Jannin. Towards responsible research in digital technology for health
861 care. 9 2021.
- 862 [33] Liang Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and
863 Hartwig Adam. Encoder-decoder with atrous separable convolution for
864 semantic image segmentation. In *Lecture Notes in Computer Science (in-*
865 *cluding subseries Lecture Notes in Artificial Intelligence and Lecture Notes*
866 *in Bioinformatics)*, volume 11211 LNCS, pages 833–851, 2018.

- 867 [34] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can Spatiotemporal
868 3D CNNs Retrace the History of 2D CNNs and ImageNet? In *Proceedings*
869 *of the IEEE Computer Society Conference on Computer Vision and Pattern*
870 *Recognition*, pages 6546–6555, 2018.
- 871 [35] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slow-
872 fast networks for video recognition. In *Proceedings of the IEEE International*
873 *Conference on Computer Vision*, volume 2019-October, pages 6201–6210, 2019.
- 874 [36] Mike Schuster and Kuldeep K Paliwal. Bidirectional Recurrent Neural
875 Networks. *IEEE TRANSACTIONS ON SIGNAL PROCESSING*, 45(11),
876 1997.
- 877 [37] Xinlei Chen and Kaiming He. Exploring Simple Siamese Representation
878 Learning. 2020.
- 879 [38] Jingru Tan, Xin Lu, Gang Zhang, Changqing Yin, and Quanquan Li.
880 Equalization Loss v2: A New Gradient Balance Approach for Long-tailed
881 Object Detection. 12 2020.
- 882 [39] Andrew Zhai and Hao Yu Wu. Classification is a Strong Baseline for Deep
883 Metric Learning. *30th British Machine Vision Conference 2019, BMVC*
884 *2019*, 11 2018.
- 885 [40] Robert Dipietro, Colin Lea, Anand Malpani, Narges Ahmidi, S. Swaroop
886 Vedula, Gyusung I. Lee, Mija R Lee, and Gregory D Hager. Recognizing
887 surgical activities with recurrent neural networks. In *Lecture Notes in*
888 *Computer Science (including subseries Lecture Notes in Artificial Intelligence*
889 *and Lecture Notes in Bioinformatics)*, volume 9900 LNCS, pages 551–558,
890 2016.
- 891 [41] Xiaojie Gao, Yueming Jin, Yonghao Long, Qi Dou, and Pheng Ann Heng.
892 Trans-SVNet: Accurate Phase Recognition from Surgical Videos via Hybrid
893 Embedding Aggregation Transformer. *Lecture Notes in Computer Science*

- 894 (including subseries *Lecture Notes in Artificial Intelligence and Lecture*
895 *Notes in Bioinformatics*), 12904 LNCS:593–603, 3 2021.
- 896 [42] François Chollet. Xception: Deep Learning with Depthwise Separable
897 Convolutions. *Proceedings - 30th IEEE Conference on Computer Vision*
898 *and Pattern Recognition, CVPR 2017*, 2017-January:1800–1807, 10 2016.
- 899 [43] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu,
900 Jianfeng Gao, and Jiawei Han. On the Variance of the Adaptive Learning
901 Rate and Beyond. 8 2019.
- 902 [44] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong
903 Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A Highly Efficient Gradient
904 Boosting Decision Tree. *Advances in Neural Information Processing Systems*,
905 30, 2017.
- 906 [45] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn,
907 and Andrew Zisserman. The Pascal Visual Object Classes (VOC) Challenge.
908 *International Journal of Computer Vision 2009 88:2*, 88(2):303–338, 9 2009.
- 909 [46] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for
910 Convolutional Neural Networks. *36th International Conference on Machine*
911 *Learning, ICML 2019*, 2019-June:10691–10700, 5 2019.
- 912 [47] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori
913 Koyama. Optuna: A Next-generation Hyperparameter Optimization Frame-
914 work. *Proceedings of the ACM SIGKDD International Conference on Knowl-*
915 *edge Discovery and Data Mining*, pages 2623–2631, 7 2019.
- 916 [48] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional
917 Networks for Biomedical Image Segmentation. *Lecture Notes in Computer*
918 *Science (including subseries Lecture Notes in Artificial Intelligence and*
919 *Lecture Notes in Bioinformatics)*, 9351:234–241, 2015.
- 920 [49] Shijie Li, Yazan Abu Farha, Yun Liu, Ming-Ming Cheng, and Juergen Gall.
921 MS-TCN++: Multi-Stage Temporal Convolutional Network for Action

- 922 Segmentation. *Proceedings of the IEEE Computer Society Conference on*
923 *Computer Vision and Pattern Recognition*, 2019-June:3570–3579, 6 2020.
- 924 [50] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Net-
925 works for Large-Scale Image Recognition. *3rd International Conference on*
926 *Learning Representations, ICLR 2015 - Conference Track Proceedings*, 9
927 2014.
- 928 [51] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluís-Miquel
929 Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean. Carbon
930 Emissions and Large Neural Network Training.
- 931 [52] Emma Strubell, Ananya Ganesh, and Andrew McCallum. Energy and Policy
932 Considerations for Deep Learning in NLP. 2019.