



HAL
open science

Late Fusion of Bayesian and Convolutional Models for Action Recognition

Camille Maurice, Francisco Madrigal, Frédéric Lerasle

► **To cite this version:**

Camille Maurice, Francisco Madrigal, Frédéric Lerasle. Late Fusion of Bayesian and Convolutional Models for Action Recognition. International Conference on Pattern Recognition (ICPR), Jan 2021, Milan (virtual), Italy. 10.1109/ICPR48806.2021.9412510 . hal-03108212

HAL Id: hal-03108212

<https://hal.science/hal-03108212>

Submitted on 13 Jan 2021

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.

Late Fusion of Bayesian and Convolutional Models for Action Recognition

Camille Maurice
LAAS-CNRS
Toulouse, France
Email: cmaurice@laas.fr

Francisco Madrigal
LAAS-CNRS
Toulouse, France
Email: jfmadrigal@laas.fr

Frédéric Lerasle
LAAS-CNRS, University Paul Sabatier
Toulouse, France
Email: lerasle@laas.fr

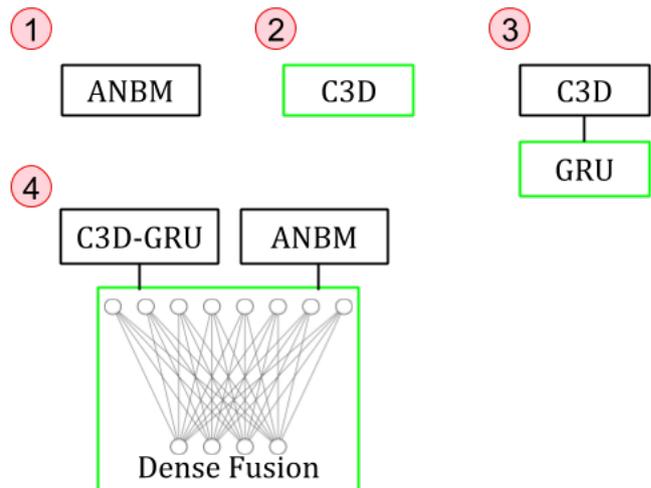
Abstract—The activities we do in our daily-life are generally carried out as a succession of atomic actions, following a logical order. During a video sequence, actions usually follow a logical order. In this paper, we propose a hybrid approach resulting from the fusion of a deep learning neural network with a Bayesian-based approach. The latter models human-object interactions and transition between actions. The key idea is to combine both approaches in the final prediction. We validate our strategy in two public datasets: CAD-120 and Watch-n-Patch. We show that our fusion approach yields performance gains in accuracy of respectively +4 percentage points (pp) and +6 pp over a baseline approach. Temporal action recognition performances are clearly improved by the fusion, especially when classes are imbalanced.

I. INTRODUCTION

The recognition of human activities is at the core of the development of many practical applications such as monitoring of domestic activities or human-robot collaboration. An activity is defined by successive time sequences of actions [1], [2] e.g.: *prepare coffee* involves the successive actions *pour in water*, *add ground coffee* and *start the machine*. On the one hand, activities performed by humans in a domestic or industrial environment can be very different, for example in the nature of the objects involved. On the other hand, the atomic actions performed may be similar in any context. Indeed, these atomic actions concern the movement of objects, their capture, or the interactions they may have with their environment. Therefore, we are interested in the recognition of atomic actions and their sequencing because higher level activities can be represented by atomic actions arranged in sequences following a logical order.

Data-driven approaches based on convolutional neural networks (CNN) adapted to the video domain with 3D convolutions allow the recognition of actions in video streams. 3D convolutional neural networks learn spatio-temporal features simultaneously. Approaches like C3D [3] obtain an accuracy of 90.4% in the action recognition dataset UCF101 [4]. However, 3D convolutions increase the size of the network and thus the number of parameters to be learned (i.e., 17M with C3D). As any CNN, they require a lot of annotated data, hence the emergence of larger annotated datasets such as NTU RGB+D [5], UCF101 [4] and Kinetic [6]. For example UCF101 contains 27 hours of videos and NTU RGB+D [5] 56880 clips. Despite these advances, action recognition is still

Fig. 1: Different individual approaches (1) (2) (3) and their fusion (4). During training, the trained layers are represented in green.



a challenge because 3D convolution networks only aggregate temporal features on video clips i.e. pre-segmented actions without temporal relations between those clips. They are taken independently and do not take into account the temporal logic in a sequence of actions.

Often, large datasets like Kinetic [6] and UCF101 [4] are created from videos collected on YouTube. The different classes are performed in radically different environments, for example *swimming* vs. *playing guitar*. However in the context of monitoring domestic activities, the actions to detect take place in a similar environment and have a temporal coherence in their sequence representing a certain activity. Recognition of sequential actions with low inter-class variance, imbalanced classes, and/or under-represented classes is still a challenge for conventional convolution networks.

Historically, probabilistic-based approaches [7], [8] propose to characterize the actions in a more explicit way through modeling the observations of the scene elements: human pose, objects and their interaction through time. These approaches usually based on probabilistic models generally offer lower

65 performance compared to convolutional networks. Neverthe- 120
66 less, they generally require less data because they also have 121
67 fewer underlying free parameters to tune. Therefore their 122
68 interpretability is less dependent on the available learning 123
69 data (e.g. less subject to over-fitting). These approaches are 124
70 relevant in the case of a small number of samples available 125
71 for training. For example, our previous Bayesian approach for 126
72 action recognition ANBM (for A New Bayesian Model [9]), 127
73 models both the interactions between objects and human- 128
74 objects through about 50 parameters. Let us note that our 129
75 ANBM approach also takes into account the transitions be- 130
76 tween different actions in order to ensure temporal consistency 131
77 throughout the sequence of actions. 132

78 Building on the observation of a possible synergy of the two 133
79 approaches, we propose a hybrid framework with a fusion at 134
80 the decision level, of a C3D [3] convolutional network and our 135
81 probabilistic ANBM [9] approach based on explicit human- 136
82 object observations. These two approaches take into account 137
83 the spatio-temporal characteristics of the different classes of 138
84 actions. Due to the large number of parameters, the C3D 139
85 network needs a lot of annotated data to be relevant since 140
86 learning is difficult in the case of under-represented classes. 141
87 The ANBM approach depends on handcrafted models and 142
88 even with a little data the prediction of under-represented 143
89 classes is possible. 144

90 Thus, our contributions are: (1) one first minor contribution 145
91 is the addition of a Gated Recurrent Unit (GRU) recurrent 146
92 layer to the C3D architecture for action recognition which 147
93 also models the temporal correlations between actions, (2) 148
94 the comparison of both approaches (ANBM and C3D-GRU) 149
95 on two public datasets CAD-120 and Watch-n-Patch, (3) 150
96 implementation and evaluation of a late fusion mechanism of 151
97 the predictions of these two approaches and comparison with 152
98 the literature. We observe a performance gain from this hybrid 153
99 approach. 154

100 The article is organized as follows. In section 2 we present 155
101 the state of the art and the context of our work. Then in 156
102 section 3 we present our hybrid approach for action detection. 157
103 A comparative study of our results is presented in section 4. 158
104 Finally, section 5 presents our conclusion and future prospects. 159

105 II. STATE OF THE ART

106 The recognition of static actions on single image can be 160
107 done by localizing certain objects in an image, i.e., Zhou 161
108 *et al.* [10] or Oquab *et al.* [11]. This kind of approach has 162
109 been popularized by the Pascal VOC 2012 challenge, where 163
110 the goal is to recognize actions in images [12]. While this is 164
111 relevant when the classes of actions to be recognized occur in 165
112 different environments, these approaches are inappropriate for 166
113 recognizing successive atomic actions occurring in a sequence 167
114 of action taking place in the same scene. It is movements and 168
115 objects involved in the execution of an action that allow it to be 169
116 discriminated, for example when opening or closing a door. 170
117 This is why we focus on approaches using spatio-temporal 171
118 information from videos in order to consider the dynamics of 172
119 gestures and objects during action classification. 173
174
175

120 Historical approaches perform dynamic action recognition 121
122 through probabilistic modeling of the observations involved. 123
124 In addition, these model-based approaches may include trajectory 125
126 models for human pose, information of the spatial configura- 127
128 tion of the objects in the scene or their affordance [13]. Li 129
130 *et al.* [7] propose the use of Gaussian mixture to recognize 131
132 different actions in the MSRAction dataset [7]. Koppula and 133
134 Saxena [8] propose the use of conditional random fields 135
136 (CRFs) to model the scene and the spatio-temporal relation- 137
138 ships that appear in CAD-120 [8]. More recently, we have 139
140 proposed a new Bayesian ANBM [9] approach based on 141
142 explicit 3D modeling of contextual features, both spatially and 143
144 temporally. These approaches rely on a smaller number of 145
146 parameters than those of C3D networks. In fact, they require 147
148 less data and are evaluated on datasets that are generally 149
150 smaller. For example MSRAction [7] contains 420 sequences 151
152 and CAD-120 [8] contains 120 videos for about 1000 clips 153
154 after segmentation of the actions. They also have the advantage 155
156 of being more interpretable than CNN approaches. 157
158

159 One of the challenges with convolutional networks is their 160
161 dependency to the amount of data available for training. 162
163 Learning their many parameters is based on the amount 164
165 of data available for training. The introduction of 3D [14] 166
167 convolution filters allows to simultaneously extract spatio- 168
169 temporal descriptors from a set of frames representing an 170
171 action, called a clip. These descriptors are appropriate for 171
172 implicitly capturing the context related to the video content. 172
173 This idea has been taken up by C3D [3] and other variants [15], 173
174 [16], [17] for action detection. Adding video clips at the 174
175 input of the network requires increasing its size compared 175
176 to its 2D CNN counterpart. C3D networks extract a global 176
177 descriptor from the clip independently of the action that took 177
178 place previously. This is particularly suitable and shows strong 178
179 results for large-scale datasets with many small clips such as 179
180 UCF101 [4] with its 13000 clips and an average duration of 180
181 7 seconds. These arrays only aggregate temporal information 181
182 over a fixed window size, typically 16 frames. This is not 182
183 suitable for recognizing actions that have temporal consistency 183
184 within their sequencing. 184

185 Hence the interest in adding a recurrent layer to a 3D- 185
186 convolutional network. Wang *et al.* [18] propose to add a Long 186
187 Short Term Memory layer (LSTM) to such a network. Also in 187
188 [19], [20] the authors propose to either add a LSTM-layer or 188
189 a GRU-layer to reinforce the temporal coherence within the 189
190 action clip and evaluate themselves on UCF101 for example. 190
191 Instead, we propose to add logical consistency in the actions 191
192 sequencing. 192

193 The fusion of C3D networks with other modalities has 193
194 already improved its performance in various challenges of 194
195 the Computer Vision community. For example, space-time 195
196 fusion [21] consists in merging an image with an optical 196
197 flow sequence that describes motion. This improves the per- 197
198 formance in comparison to a C3D network alone, which seeks 198
199 to simultaneously extract temporal and spatial features at the 199
200 3D convolution layers. There are also methods that propose 200
201 a fusion of different features of different nature such as 201
202

176 audio and video [22]. These different approaches show the
 177 advantages of using a fusion mechanism to increase overall
 178 performance. However, this gain is achieved at the expense
 179 of the amount of data required for training. The addition of
 180 more modalities increases the number of parameters to be
 181 learned for the convolution network. This has two effects:
 182 first it required the existence of a such dataset, and second it
 183 increases the training time. A late fusion is proposed by [23]
 184 for pose attention in RGB videos.

185 We propose to merge two spatio-temporal approaches, one
 186 based on context modeling via learning such as C3D [3] and
 187 our ANBM [9] approach based on Bayesian models and 3D
 188 human and objects observations of the scene. This fusion is
 189 not done at the feature level but later at their predictions level
 190 towards the same layer. We propose to merge them using
 191 a fully connected layer, i.e. dense layer. Only a few works
 192 study the late fusion of two classes of approaches that a priori
 193 complement each other and the gains that this can yield.

194 Public datasets such as Watch-n-Patch [24] and CAD-
 195 120 [8] allow to evaluate the recognition of atomic actions.
 196 These datasets offer approximately 20-seconds long videos
 197 in which different atomic actions are annotated. The actions
 198 follow each other in a logical order, for example we cannot
 199 move an object that has not been previously captured. In these
 200 datasets, the sequences of actions are more or less correlated.
 201 Moreover some classes are under-represented in these datasets,
 202 which is generally a lock for C3D learning.

203 III. PROPOSED APPROACH

204 In this section we describe the proposed architecture for the
 205 fusion of probabilities predicted by the ANBM [9] Bayesian
 206 approach with those of the modified C3D [3] network. We
 207 recall our previous ANBM approach in Section III-A, and
 208 then briefly describe the C3D network in section III-B and
 209 its modification (C3D-GRU) in Section III-C. Section III-D
 210 details the proposed late fusion strategy.

211 A. Bayesian Approach With Human-Object Observations

212 This approach [9] is based on the following insights:
 213 human pose, human-object and object-to-object interactions,
 214 performed during the execution of an action, provide spatio-
 215 temporal information that allows the recognition of the on-
 216 going action. Moreover, it considers temporal information such
 217 as transition between actions during a sequence. We have
 218 modeled these observations in order to be able to estimate,
 219 at each time of the video, the probabilities of each considered
 220 actions.

221 All the elements of the scene are first localized in the
 222 image plane by 2D state-of-the-art detectors one for human
 223 pose estimation an another for the objects. Then they are
 224 modeled in 3D space using RGB-D sensor (e.g. Kinect)
 225 calibration data. The detection of the human pose in the image
 226 is based on OpenPose [25], which is trained on MSCOCO
 227 Keypoints Challenge [26]. We use Single Shot Multi-Box
 228 Detector (SSD) [27] to recognize objects, which is trained
 229 with the MSCOCO dataset [26].

Each action a is associated to a model. Let $A =$
 $\{a^1, a^2, \dots, a^N\}$ be the set of N actions. The joint observation
 of the human pose s_t and the set of objects Ω_t is described at
 time t by $O_t = \{s_t, \Omega_t\}$ where $\Omega_t = \{\omega^1, \omega^2, \dots, \omega^{Card(\Omega)}\}$
 with $Card(\Omega)$ being the number of objects in the scene. The
 inference is performed on a sliding window of T frames, so
 that this approach does not require video clips segmentation
 beforehand, and ensure temporal consistency of the observa-
 tions. We model the *a posteriori* probability of the actions
 given the observations as follows:

$$p(a_{0:T}|O_{0:T}) \propto \prod_{t=0}^T p(O_t|a_t) \prod_{t=1}^T p(a_t|a_{t-1}). \quad (1)$$

230 Where $p(O_t|a_t)$ is the likelihood of the observation given the
 231 action a_t . The term $p(a_t|a_{t-1})$ characterizes the probabilities
 232 of transitions between two successive actions. All the observa-
 233 tions of the scene in this approach are modelled in 3D.
 234 Objects and pose 2D coordinates are projected onto the 3D
 235 space thanks to the sensor calibration data. It allows ANBM
 236 to be more robust to changes of point of view than an approach
 237 based solely on 2D spatial characteristics. We invite the reader
 238 to consult [9] the paper for more in-depth details.

239 B. 3D convolution network: C3D

240 C3D [3] is a deep learning network that takes into account,
 241 in addition to images, a third dimension corresponding to time.
 242 The architecture includes $3 \times 3 \times 3$ convolution filters, followed
 243 by $2 \times 2 \times 2$ pooling layers. The introduction of 3D convolution
 244 filters allows to learn spatio-temporal descriptors from a video
 245 stream.

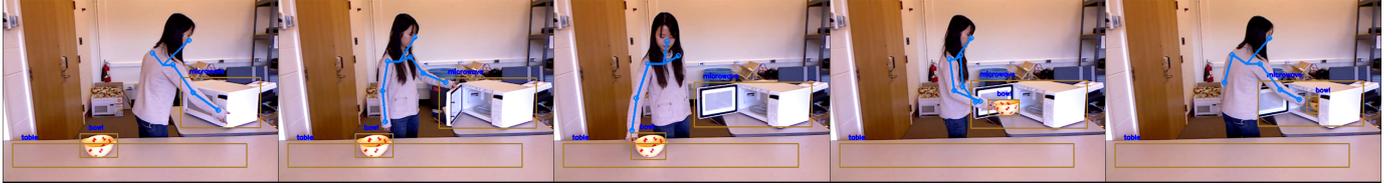
246 On the one hand, they provide a compact (4096) description
 247 of a video stream of size $H \times W \times C \times L$. With L the length
 248 of the video clip containing the action, usually 16 frames.
 249 These networks are able to learn those spatio-temporal descrip-
 250 tors implicitly. Like other networks they perform end-to-end
 251 learning without expert information (unlike any probabilistic
 252 approach e.g. ANBM).

253 On the other hand, given the millions of parameters to be
 254 learned, poorly represented classes are hardly well recognized
 255 and it is more difficult to predict them correctly. The action
 256 must be sampled on 16 frames in the original implementation,
 257 of course it is possible to enlarge this time window but it
 258 requires more memory. Tran *et al* [3] also offer a sliding
 259 window system of descriptor averaging. However, in all cases,
 260 C3D requires a pre-segmentation of actions in sequences,
 261 which does not make it a suitable method for online action
 262 recognition. Moreover, there are no mechanisms to take into
 263 account the temporal context in which the video clip is inserted
 264 during training. Therefore there is no consideration of the
 265 previous action.

266 C. Adding a recurrent layer: C3D-GRU

267 In order to compensate the lack of a mechanism that ensure
 268 the temporal consistency along the sequence, across the video-
 269 clips. We propose to take into account the previous action in
 270 the detection of the current action by adding a recurrent layer.

Fig. 2: An action sequence from CAD-120 [8] dataset: actor 1, video 2305260828, action *microwaving-food*. From left to right : reach, open, reach, move, place. In blue: human pose detected by OpenPose. In yellow: objects detected by SSD.



271 Once we trained C3D, we retrieve its weights, freeze them
 272 and add a recurrent GRU-type layer. C3D is trained with data
 273 augmentation that is not able to perform in the same manner
 274 for the GRU-type layer. Indeed we need to preserve temporal
 275 coherence and segments to train in logic order.

276 Then we adapt the GRU layer to take into account two
 277 successive clips corresponding to two different, but successive,
 278 actions. We do not re-train the whole C3D network but we only
 279 perform a fine-tuning at the level of the last layers. To illustrate
 280 the importance of the nature of the previous actions we notice
 281 that among all the possible transitions between any two pairs
 282 of actions in Watch-n-Patch [8], only about 20 % are actually
 283 occurring. By adding this extra constraint while training the
 284 GRU-layer, we hope to reduce the number of false positive
 285 detections of some classes. This strategy is illustrated in Fig. 1,
 286 number 3. We call this approach C3D-GRU afterwards.

287 D. Late fusion with a dense layer

288 We therefore have two approaches to predict actions from
 289 video clips based on spatio-temporal data, explicitly with
 290 ANBM and implicitly with C3D-GRU. Both approaches also
 291 consider the existing transitions between two successive actions.
 292 On the one hand with ANBM we have modeled each
 293 action, on the other hand C3D-GRU learns from the datasets,
 294 whose classes are not equally distributed. Indeed in the detection
 295 of atomic actions, some actions are found more frequently.
 296 For example the displacement of an object (*moving*) represents
 297 34% of the actions of CAD-120, it mandatory occurs before
 298 many different actions such as *to drink* because we need *to*
 299 *move* the bottle before. We propose a fusion of their respective
 300 predictions. Both approaches estimate probabilities for each
 301 class. We have one for ANBM and for C3D-GRU we have a
 302 vector corresponding to the output of the soft-max layer.

303 We propose a strategy that takes as input video clips that
 304 are processed through the ANBM approach and also through
 305 the C3D-GRU network described above, whose C3D layers
 306 weights are frozen. We thus obtain two prediction vectors
 307 for each of the approaches that are later concatenated. This
 308 concatenation is connected to a dense layer of the same size
 309 as the number of classes, as shown with only $N = 4$ classes
 310 as example in Fig 1, number 4. So there are only $N^2 + N$
 311 parameters to learn (N^2 weights related to the dense layer and
 312 N bias related to activation). This interconnection enables to
 313 take the advantage of both approaches in the final decision.
 314 We call this approach C3D-GRU-DF thereafter.

315 IV. EXPERIMENTS AND RESULTS

316 A. Public Datasets

317 Let us recall that we propose an initial approach to detect
 318 actions in [9]. This online approach is able to detect actions
 319 sequences of from a video stream and to manage transitions
 320 between actions. We wish to take advantage of this asset, so
 321 we evaluate ourselves on two public datasets which contain
 322 such action sequences: CAD-120 [8] and Watch-n-Patch [24].

323 a) *CAD-120*: The CAD-120 [8] dataset consists of 120
 324 videos with RGB-D channels, played by 4 actors. It contains
 325 10 daily life activities (preparing a bowl of cereal, taking
 326 medication...). These activities involve 10 actions: reaching,
 327 moving, pouring, eating, drinking, placing, opening, closing,
 328 null. Here, each video represents an activity as defined in the
 329 section I. The inequitable distribution of actions, expressed
 330 by the corresponding percentage of frames, is described in
 331 Tab. III. An illustration of this dataset is presented in Fig. 2.

332 b) *Watch-n-Patch*: The office environment consists of
 333 196 videos recorded in 8 different offices. There are 10
 334 annotated actions: read, walk, leave-office, fetch-book, put-
 335 back-book, put-down-item, pick-up-item, play-computer, turn-
 336 on computer, turn-off computer. Here again some actions are
 337 dependent on the action that takes place previously, e.g. to play
 338 the computer, the screen must be turned on. Action classes are
 339 not equally distributed as shown in Tab. III.

TABLE III: Detail of class distribution within datasets and the number of clips.

Dataset	Number of clips	Distribution (% per class)
CAD-120	1149	[23,30,3,3,3,15,4,3,1,14]
Watch-n-Patch	1148	[12,16,21,6,4,14,9,9,5,3]

340 B. System Evaluation

341 a) *Managing ANBM's Predictions*: We record the pre-
 342 diction probabilities of ANBM at each frame, then we take
 343 their averages over the duration of each action to assign a
 344 class to each video clip representing an action.

345 b) *Pre-processing for C3D*: We keep the original settings
 346 of the publication [3] for the input image size by setting it to
 347 112 x 112 pixels. The video clips are cropped around the
 348 enlarged bounding box containing the actor and objects in the
 349 action context. This bounding box is detected using the human
 350 pose inferred by OpenPose [25]. This allows the network to

TABLE I: Results of our different variants on Watch-n-Patch. Performance metrics considered are macro-accuracy (M) and micro-accuracy (μ).

Architecture	Sample 0		Sample 1		Sample 2		Sample 3		Mean		Standard Deviation	
	μ	M	μ	M								
1 - ANBM	0.78	0.79	0.73	0.74	0.76	0.77	0.75	0.75	0.76	0.76	0.02	0.02
2 - C3D	0.72	0.65	0.73	0.64	0.75	0.69	0.74	0.64	0.74	0.66	0.01	0.02
3 - C3D-GRU	0.89	0.87	0.86	0.77	0.85	0.77	0.89	0.84	0.87	0.81	0.02	0.05
4 - C3D-GRU-ANBM-DF	0.94	0.91	0.93	0.90	0.93	0.91	0.93	0.89	0.93	0.90	0.001	0.01

TABLE II: Results of our different variants on CAD-120. Performance metrics considered are macro-accuracy (M) and micro-accuracy (μ).

Architecture	Actor 1		Actor 2		Actor 3		Actor 4		Mean		Standard Deviation	
	μ	M	μ	M								
1 - ANBM	0.84	0.77	0.78	0.81	0.82	0.76	0.82	0.77	0.82	0.78	0.03	0.02
2 - C3D	0.58	0.45	0.70	0.61	0.64	0.57	0.56	0.35	0.62	0.50	0.06	0.12
3 - C3D-GRU	0.61	0.49	0.76	0.73	0.66	0.60	0.60	0.45	0.66	0.57	0.07	0.13
4 - C3D-GRU-ANBM-DF	0.86	0.80	0.89	0.91	0.84	0.82	0.83	0.79	0.86	0.83	0.03	0.05

focus its attention on the area where the activity is taking place. The C3D network takes an action sequence of fixed size: 16 frames. In practice, since we consider atomic actions, which are relatively short, we do not use a sliding window on the sequences but rather simply re-sample the sequences.

c) Training: The network weights are trained using a stochastic gradient descent on mini-batches of size 16 with a momentum of 0.9. We initialize the learning rate to 0.01 and it decreases over time. The training is done on a GeForce GTX 1080 Ti graphics card. We use the cross-entropy categorical loss function.

d) Testing: The performance of our hybrid approach is evaluated according to the principle of *k-fold* cross-validation where the *k*-folds form a partition of the dataset (with *k* = 4). Each fold is used exactly once as a validation set during training. In the CAD-120 dataset there are four actors and each fold is associated with one actor. In Watch-n-Patch, the original publication [24] provides one test and training sets, we generate 3 more folds while keeping the actions in the same sequence within the same fold. We obtain the final prediction at the last activation layer, softmax, present in variants 2,3 and 4 described in section III and illustrated in Fig. 1.

C. Metrics for Evaluation

We evaluate the different variants proposed in section III with two metrics. The first one is the accuracy, later called micro-accuracy (μ), which is defined as follows:

$$\mu\text{-accuracy} = \frac{\text{number of correct predictions}}{\text{total number of predictions}}. \quad (2)$$

This measures the ratio of correctly recognized actions to the total number of actions to recognize. In contrast, the second

metric called macro-accuracy (M) measures the average of the accuracy for each class. The accuracy of each class is calculated and the macro-accuracy is the average of these accuracies. Macro-accuracy gives the same weight to each class, regardless of the number of samples the class has in the dataset. This makes possible to see if only the most represented classes are correctly recognized or if globally all the classes, including the under-represented ones, are correctly recognized. These two metrics are complementary in performance evaluation for datasets with imbalanced classes.

D. Results and discussion

We first compare the individual results of C3D, C3D-GRU and ANBM variants described in section III before evaluating their late fusion.

Here we evaluate the contribution of the GRU recurrent layer at the output of C3D to take into account the temporal logic between actions (C3D-GRU). According to Tab. I in Watch-n-Patch we observe, on average, a gain in micro-accuracy of +13 percentage points (cf. lines 2 and 3). Regarding CAD-120 dataset, we observe on Tab. II (cf. lines 2 and 3) a gain in micro-accuracy of +4 percentage points thanks to the addition of a GRU layer to the C3D network compared to C3D alone. Looking in detail at the different confusion matrices obtained on Watch-n-Patch on Fig. 5 and 6, we see that classes that benefit the most are the following: *put-back-book*, *put-down-item* and *take-item*. Indeed the action *put-back-book* is often preceded by the action *read*. When the previous action is labeled as *read*, it reduces and conditions the choice of the following possibilities. The action *put-down-item* is often preceded by action *walk*. Indeed, in Watch-n-Patch it

406 is a common scenario for a person to walk into the office
 407 and put his or her phone on the table. The gains on CAD-
 408 120 are more modest because for the recurrent layer to bring
 409 information, the frozen C3D network must have learned to
 410 recognize classes with a sufficient accuracy.

411 The C3D-GRU network therefore outperforms C3D and
 412 now we are comparing it with our ANBM approach before
 413 evaluating their fusion. On the Watch-n-Patch dataset, C3D-
 414 GRU has a better micro-accuracy than ANBM (+11 percentage
 415 points pp) but the improvement of the macro-accuracy is less
 416 important (+5 pp), cf. lines 1 and 3 of Tab.I. As it can be seen
 417 on the confusion matrix on Fig. 6, the best detected actions
 418 by C3D-GRU are *read*, *walk* and *leave-office* with scores of
 419 1, 0.98, and 0.97% respectively. These actions represent 49%
 420 of the data (cf. Tab. III of the dataset) and contribute more
 421 to the micro-accuracy than, for example, *turn-off-computer*
 422 which represents only 3%. The confusion matrix of Fig. 4
 423 shows us that the ANBM approach outperforms C3D-GRU
 424 on 3 classes: *play-computer*, *turn-on-computer* and *turn-off-*
 425 *computer*. Both approaches perform best on different classes,
 426 but the nature of false positives also varies. As we can see
 427 from the confusion matrices in Figs. 4 and 6, both approaches
 428 have similar performances for the action *fetch-book* (0.71%
 429 for ANBM and 0.81% for C3D-GRU) but the errors differ.
 430 Indeed ANBM sometimes detects *reach* while C3D-GRU
 431 detects *place* instead of *fetch-book* On CAD-120 the C3D-
 432 GRU network distinguishes better *reaching* and *placing* than
 433 ANBM, these two actions represent 45% of the dataset.

434 Thus, both datasets C3D-GRU and ANBM bring perfor-
 435 mances that complement each other. Giving best performances
 436 on different classes and different false positive sources of error
 437 may be one reason why the fusion using a fully connected
 438 layer may capture more information than a simple average
 439 of the two outputs. Here we evaluate the benefits that can
 440 be derived from their fusion. On CAD-120, the ability of the
 441 C3D-GRU network to distinguish *reach* and *place* from other
 442 classes allows, when merging both approaches, a gain of +4
 443 percentage points in micro-accuracy, see Tab. II. The fusion
 444 of C3D-GRU and ANBM improves the recognition of every
 445 actions on Watch-n-Patch except the action *walk* (1) which
 446 drops from 0.98 with C3D-GRU to 0.97 as well as actions
 447 *play-computer* (7) and *turn-off-computer* (9) that also drop
 448 by 1% with C3D-GRU-ANBM-DF fusion as shown by the
 449 confusion matrices on Figs. 4 6 and 7. Overall, predictions
 450 fusion increases the micro-accuracy by +6 percentage points
 451 with respect to C3D-GRU and by +17 percentage points with
 452 respect to ANBM. In the fusion, for actions involving a
 453 computer the performances of ANBM are favoured over those
 454 of C3D-GRU. The fact that both approaches complement each
 455 other is also well exploited when they individually present
 456 similar performances for a same class. For example, with the
 457 fusion approach the action *fetch-book* reaches 0.94 whereas
 458 with C3D-GRU and ANBM this action was correctly predicted
 459 in respectively 0.81 and 0.77.

460 Here we propose to evaluate the robustness of this fusion
 461 approach on Watch-n-Patch by smoothing or further degrad-

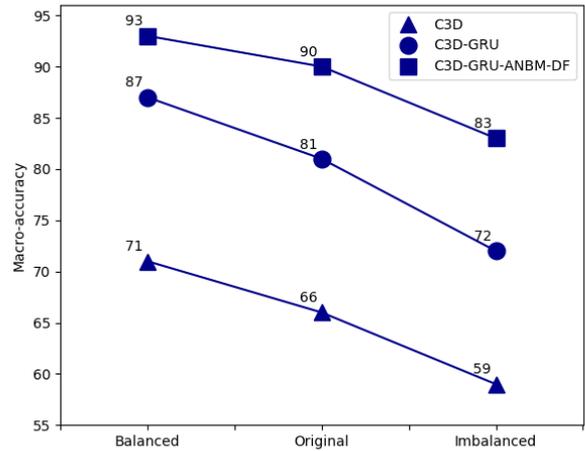


Fig. 3: Macro-accuracy with respect to Watch-n-Patch dataset with classes synthetically augmented in order to decrease or increase the class imbalance.

TABLE IV: Comparison to the literature. Action recognition accuracy on two public datasets: CAD-120 and Watch-n-Patch.

Dataset	Approaches	Accuracy
CAD-120	GEPHAPP [28]	79.4
	ANBM [9]	82.2
	GPNN [29]	87.3
	Ours	86.1
Watch-n-Patch	CaTM [24]	32.9
	WBTM [30]	35.2
	PoT [31]	49.93
	ANBM [9]	76.4
	GEPHAPP [28]	84.8
	Ours	93.0

462 ing the class imbalance within the dataset. We synthetically
 463 augment or degrade the dataset and we re-train the networks
 464 C3D, C3D-GRU, C3D-GRU-ANBM-DF to obtain the results
 465 presented in Fig. 3. We observe that C3D training is sensitive
 466 to the number of samples in the training. We also observe the
 467 dependency of C3D-GRU on the result of C3D. Indeed C3D-
 468 GRU performance drops faster than C3D, because to capture
 469 temporal coherence, the previous action must be well detected.
 470 When classes are strongly imbalanced, C3D detects poorly
 471 some actions and some temporal transitions between action
 472 are not modeled. Overall, as expected we note that the fusion
 473 of C3D-GRU-ANBM-DF resist more to the degradation of the
 474 samples, with a slightly less important slope value.

475 As shown in Tab. IV, our hybrid fusion strategy allows
 476 us to improve our previous performance, while still having
 477 near or better than the state of the art performances. We
 478 select recent state-of-the-art benchmark approaches and if
 479 possible that are evaluated on the same two datasets such as

Fig. 4: Confusion matrix of ANBM (original test set from Watch-n-Patch). Predictions are on columns are ground truth on rows. [0 - read ; 1 - walk ; 2 - leave-office ; 3 - fetch-book ; 4 - put-back-book ; 5 - put-down-item ; 6 - take-item ; 7 - play-computer ; 8 - turn-on-computer ; 9 - turn-off-computer]

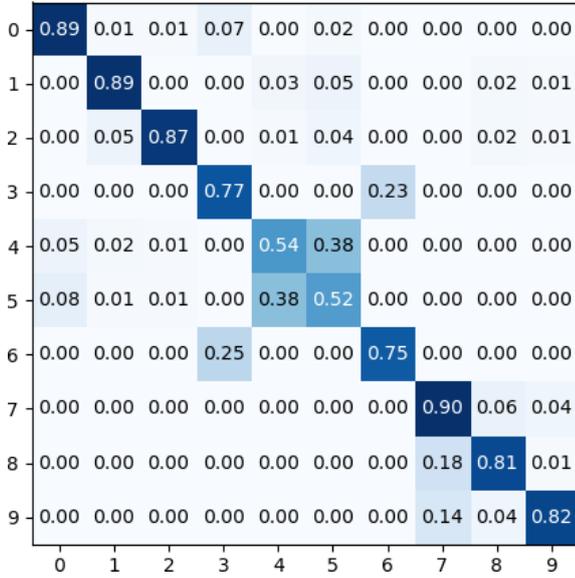
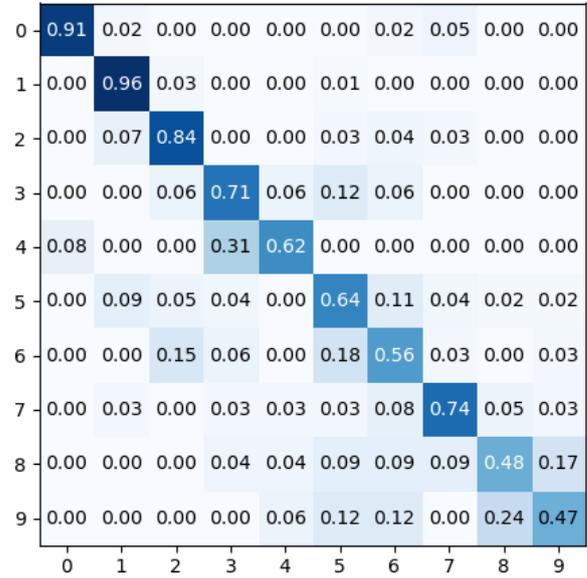


Fig. 5: Confusion matrix of C3D (original test set from Watch-n-Patch). Predictions are on columns are ground truth on rows. [0 - read ; 1 - walk ; 2 - leave-office ; 3 - fetch-book ; 4 - put-back-book ; 5 - put-down-item ; 6 - take-item ; 7 - play-computer ; 8 - turn-on-computer ; 9 - turn-off-computer]



480 Qi *et al.* [28]. The two approaches considered in our hybrid
 481 fusion strategy take into account the transitions between two
 482 successive actions. In CAD-120, the action *moving* precedes
 483 almost all the others, which is not very informative. We may
 484 consider to take into account more transitions. It also shows
 485 that our merging strategy allows us to surpass the state of
 486 the art in action recognition on the Watch-n-Patch dataset
 487 by improving action recognition by +8.2 percentage points
 488 compared to the approach proposed by Qi *et al.* [28].

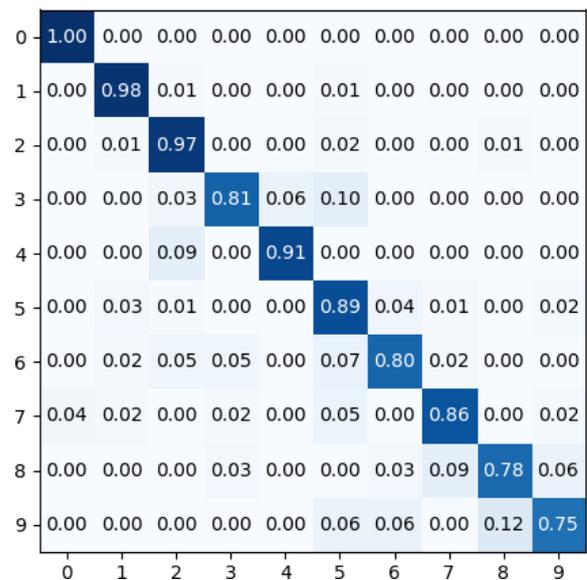
489 A video illustrates our results on video sequences of both
 490 datasets is available at the address indicated in the footnote ¹.

491 V. CONCLUSION AND FUTURE WORK

492 In this paper we have compared different approaches for
 493 action detection and proposed the addition of a recurrent layer
 494 to C3D to benefit from the temporal relationships between
 495 actions. We explored a way to merge at the decision level
 496 of data driven and Bayesian-based approaches for action
 497 recognition using a dense layer. We experimented with two
 498 datasets in the literature presenting an imbalance between their
 499 classes, and we show gains in accuracy that are even more
 500 significant when the approaches complement each other.

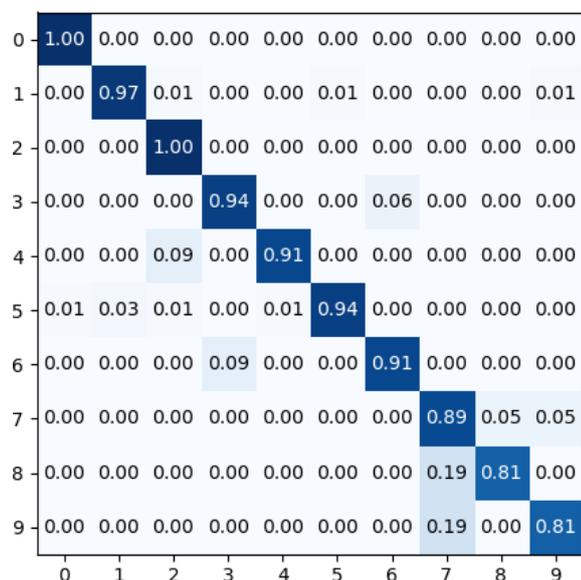
501 In the perspectives we plan to evaluate our merging ap-
 502 proach on the detection of high-level activities composed by
 503 the succession of atomic actions. In the future we also plan
 504 to further investigate deep learning architectures for automatic
 505 action segmentation in order to deal with untrimmed video
 506 data.

Fig. 6: Confusion matrix of C3D-GRU (original test set from Watch-n-Patch). Predictions are on columns are ground truth on rows. [0 - read ; 1 - walk ; 2 - leave-office ; 3 - fetch-book ; 4 - put-back-book ; 5 - put-down-item ; 6 - take-item ; 7 - play-computer ; 8 - turn-on-computer ; 9 - turn-off-computer]



¹<https://youtu.be/7txCiHx3OWA>

Fig. 7: Confusion matrix of our fusion approach C3D-GRU-ANBM-DF (original test set from Watch-n-Patch). Predictions are on columns are ground truth on rows. [0 - read ; 1 - walk ; 2 - leave-office ; 3 - fetch-book ; 4 - put-back-book ; 5 - put-down-item ; 6 - take-item ; 7 - play-computer ; 8 - turn-on-computer ; 9 - turn-off-computer]



ACKNOWLEDGEMENT

This work has been partially supported by Bpifrance within the French Project LinTO and funded by the French government under the Investments for the Future Program (PIA3).

REFERENCES

[1] P. Turaga, R. Chellappa, V. S. Subrahmanian, and O. Udrea, "Machine recognition of human activities: A survey," *IEEE Transactions on Circuits and Systems for Video technology*, vol. 18, no. 11, pp. 1473–1488, 2008.

[2] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Computer vision and image understanding*, vol. 104, no. 2-3, pp. 90–126, 2006.

[3] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3d convolutional networks," in *Proceedings of the IEEE international conference on computer vision*, pp. 4489–4497, 2015.

[4] K. Soomro, A. R. Zamir, and M. Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *arXiv preprint arXiv:1212.0402*, 2012.

[5] A. Shahroudy, J. Liu, T.-T. Ng, and G. Wang, "Ntu rgb+d: A large scale dataset for 3d human activity analysis," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1010–1019, 2016.

[6] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

[7] W. Li, Z. Zhang, and Z. Liu, "Action recognition based on a bag of 3d points," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, pp. 9–14, IEEE, 2010.

[8] H. S. Koppula, R. Gupta, and A. Saxena, "Learning human activities and object affordances from rgb-d videos," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 951–970, 2013.

[9] C. Maurice, F. Madrigal, A. Monin, and F. Lerasle, "A new bayesian modeling for 3d human-object action recognition," in *2019 16th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pp. 1–8, IEEE, 2019.

[10] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2921–2929, 2016.

[11] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1717–1724, 2014.

[12] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International journal of computer vision*, vol. 88, no. 2, pp. 303–338, 2010.

[13] A. Gupta and L. S. Davis, "Objects in action: An approach for combining action understanding and object perception," in *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2007.

[14] S. Ji, W. Xu, M. Yang, and K. Yu, "3d convolutional neural networks for human action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 1, pp. 221–231, 2012.

[15] D. Tran, J. Ray, Z. Shou, S.-F. Chang, and M. Paluri, "Convnet architecture search for spatiotemporal feature learning," *arXiv preprint arXiv:1708.05038*, 2017.

[16] G. Varol, I. Laptev, and C. Schmid, "Long-term temporal convolutions for action recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 6, pp. 1510–1517, 2017.

[17] K. Liu, W. Liu, C. Gan, M. Tan, and H. Ma, "T-c3d: temporal convolutional 3d network for real-time action recognition," in *Thirty-second AAAI conference on artificial intelligence*, 2018.

[18] X. Wang, L. Gao, J. Song, and H. Shen, "Beyond frame-level cnn: saliency-aware 3-d cnn with lstm for video action recognition," *IEEE Signal Processing Letters*, vol. 24, no. 4, pp. 510–514, 2016.

[19] N. Lu, Y. Wu, L. Feng, and J. Song, "Deep learning for fall detection: Three-dimensional cnn combined with lstm on video kinematic data," *IEEE journal of biomedical and health informatics*, vol. 23, no. 1, pp. 314–323, 2018.

[20] G. Yao, X. Liu, and T. Lei, "Action recognition with 3d convnet-gru architecture," in *ICRA*, pp. 208–213, 2018.

[21] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1933–1941, 2016.

[22] Y. Fan, X. Lu, D. Li, and Y. Liu, "Video-based emotion recognition using cnn-rnn and c3d hybrid networks," in *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pp. 445–450, 2016.

[23] F. Baradel, C. Wolf, and J. Mille, "Human activity recognition with pose-driven attention to rgb," 2018.

[24] C. Wu, J. Zhang, S. Savarese, and A. Saxena, "Watch-n-patch: Unsupervised understanding of actions and relations," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4362–4370, 2015.

[25] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7291–7299, 2017.

[26] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[27] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37, Springer, 2016.

[28] S. Qi, B. Jia, S. Huang, P. Wei, and S.-C. Zhu, "A generalized earley parser for human activity parsing and prediction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[29] S. Qi, W. Wang, B. Jia, J. Shen, and S.-C. Zhu, "Learning human-object interactions by graph parsing neural networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 401–417, 2018.

[30] C. Wu, J. Zhang, B. Selman, S. Savarese, and A. Saxena, "Watch-bot: Unsupervised learning for reminding humans of forgotten actions," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2479–2486, IEEE, 2016.

[31] M. S. Ryoo, B. Rothrock, and L. Matthies, "Pooled motion features for first-person videos," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 896–904, 2015.

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615