Can Deep Learning Recognize Subtle Human Activities?

Vincent Jacquot
École Polytechnique
Fédérale de Lausanne
jacquot.vinc@gmail.com

Zhuofan Ying
University of Science and
Technology of China
zhuofanying@gmail.com

Gabriel Kreiman
Center for Brains, Minds
and Machines, Boston, MA
gabriel.kreiman@tch.harvard.edu

Abstract

Deep Learning has driven recent and exciting progress in computer vision, instilling the belief that these algorithms could solve any visual task. Yet, datasets commonly used to train and test computer vision algorithms have pervasive confounding factors. Such biases make it difficult to truly estimate the performance of those algorithms and how well computer vision models can extrapolate outside the distribution in which they were trained. In this work, we propose a new action classification challenge that is performed well by humans, but poorly by state-of-the-art Deep Learning models. As a proof-of-principle, we consider three exemplary tasks: drinking, reading, and sitting. The best accuracies reached using state-of-the-art computer vision models were 61.7%, 62.8%, and 76.8%, respectively, while human participants scored above 90% accuracy on the three tasks. We propose a rigorous method to reduce confounds when creating datasets, and when comparing human versus computer vision performance. Source code and datasets are publicly available[1].

1. Introduction

Deep convolutional neural networks have radically accelerated progress in visual object recognition, with impressive performance on datasets such as ImageNet [31], achieving top-5 error of 16.4 % in 2012 [20], down to 1.8% in 2019 [44]. Similar progress has been observed in other domains such as action recognition, with an error rate of 1.8% [6] in the UCF101 dataset [35].

Such impressive feats have also been accompanied by vigorous discussions to better understand what the networks learn and how they classify the images [46, 24, 32, 28, 18]. In addition to showcasing algorithmic successes, systematically understanding the networks' limitations will help us develop better and more stringent datasets to stress test
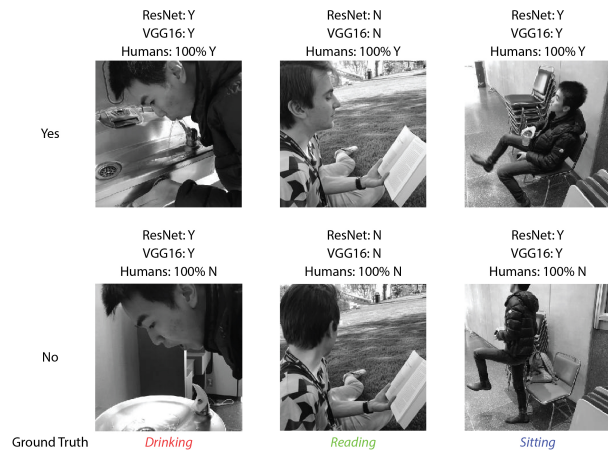

Figure 1. **Example images from our dataset (Group 2, controlled set)**. Left to right: *drinking*, *reading*, and *sitting*. Top: positive images. Bottom: negative images. Above each image, classification output for ResNet, VGG16, and human psychophysics measurements (see text for details). The models misclassified the middle top, bottom left, and bottom right pictures, whereas humans correctly classified all the pictures. See also **Fig. S4**.

models and develop better ones. For example, in the UCF101 dataset, algorithms can rely exclusively on the background color to classify human activities well above chance levels. For example, "sky diving" typically correlates with blue pixels (the sky), whereas 'baseball pitch' correlates with green pixels (the field).

As an illustration of how to rigorously test state-of-the-art models, and how to build controlled datasets, we focus on action recognition from individual frames. We study three human behaviors: whether a person is *drinking* or not, *reading* or not, and *sitting* or not (**Figure 1**, **Fig. S4**). Each of these actions is considered independently in a binary classification task. We first describe how we built a controlled dataset, next we demonstrate that humans can rapidly solve these tasks, and finally we show that these simple binary questions challenge current systems, and introduce initial thoughts on how such tasks could be solved.

---

[1] https://github.com/kreimanlab/
DeepLearning-vs-HighLevelVision

2. Related Work

Object detection. Large datasets for object detection have played a critical role in recent progress in computer vision. The success of Krizhevsky *et al.* [20] on ImageNet [31] triggered the development of powerful algorithms [44, 41, 25], and multiple datasets such as COCO [23].

Action recognition. In a similar fashion, multiple datasets have been developed to train algorithms to recognize actions, including the MPII Human Pose [2], COCO keypoints [23], Leeds Sports Pose [16], UCF101 action [35], and Posetrack [1] datasets. These datasets led to the current state-of-the art models for human pose estimation [40, 39, 42, 5, 29, 4].

Current challenges and possible approaches. There has been significant progress in developing enhanced algorithms for recognition combining region proposal [11, 10, 9, 14, 12], distinction between foreground/background and other scene elements [22, 30, 17, 12], and interactions between image parts [13].

Despite enormous progress triggered by these datasets, there exist strong low-level biases that correlate with the labels. For example, the work of Xiao *et al.* showed that a simple architecture, combining ResNet with several deconvolution layers, reached the top accuracy of 73.7% mAP in human pose estimation and tracking [43]. This type of challenge is particularly notable in datasets like UCF-101: extracting *merely the first frame* of each video, converting it to grayscale, and using an SVM classifier with a linear kernel, it is possible to obtain performance levels well above chance in "action recognition". To capitalize on the power of current algorithms, and to push the development of even better ones, it is essential to stress test computer vision systems with sufficiently well-controlled datasets that cannot be solved by simple heuristics. Here we focus on the problem of action recognition from static images and provide intuitions about the development of a well-controlled dataset to challenge computational algorithms.

3. Building a Controlled Dataset

We sought to create a dataset to challenge and improve current recognition algorithms, focusing on action recognition from single frames in three examples: drinking, reading, and sitting. Datasets that involve discriminating among completely different actions (as in UCF-101, [35]), often incorporate extensive background information that can help solve the discrimination problem by capitalizing on basic image heuristics (as noted in the introduction for the example of skydiving versus baseball pitch). Therefore, here we take a different approach and focus on binary tasks of the form: is the person drinking or not, reading or not, sitting or not. We do *not* compare drinking to reading to sitting (i.e., vertical and not horizontal comparisons in **Figure 1**).

3.1. Dataset collection

The images originated from two sources: (Group 1) Photographs manually downloaded from open source materials on the Internet; (Group 2) New custom photographs taken by investigators in our lab.

Despite our best efforts, we quickly realized that Group 1 (internet images) contained strong biases: even an SVM with a linear kernel applied to the image pixels could classify images with higher-than-chance accuracy. Consequently, we decided to take our own photographs (Group 2, controlled set, **Figure 1**, **Fig. S4**). Special care was taken to avoid biases when taking pictures. Whenever we took a photo representing a behavior in a certain setting (e.g., person A drinking from a cup in location L), we also took a companion photo of the opposite behavior in the same setting (person A holding the same cup in location L but *not* drinking). Examples of these image pairs for each behavior are shown in **Figure 1**. The opposite behavior could be a slight change, for example the same picture with and without water in the case of *drinking*, or changing the direction of gaze for *reading*, or changing body posture for *sitting*. This procedure ensured that the differences between the two classes could not be readily ascribed to low-level properties associated with the two labels. We reasoned that these differences between the *yes* and *no* classes would make the classification task difficult for current algorithms, while still being solvable by humans. We conjectured that these subtle, but critical, differences, highlight the key ingredients of what it means for an algorithm to be able to truly recognize an action.

The original number of images in the *drinking*, *reading* and *sitting* datasets were 4,121, 3,071 and 3,684, respectively. These datasets were then split into *yes* and *no* classes according to the labelling procedure described in Section 3.2. About 85% of each dataset consisted of our own photographs (Group 2), while the rest was from the Internet (Group 1). All images were converted to grayscale and resized to 256-by-256 pixels (except in **Fig. S1** and **Fig. S2** which show results for RGB images).

3.2. Labelling images

We created ground truth labels for each image by asking 3 participants to assign each image to a *yes* or *no* class for each action. The participants were given simple guidelines to define each action: *drinking* (liquid in mouth), *reading* (gaze towards text), and *sitting* (buttocks on support). In contrast to the psychophysics tests in Section 4, here the 3 participants had no time constraint to provide labels. We only kept an image if all the participants agreed on the class label.
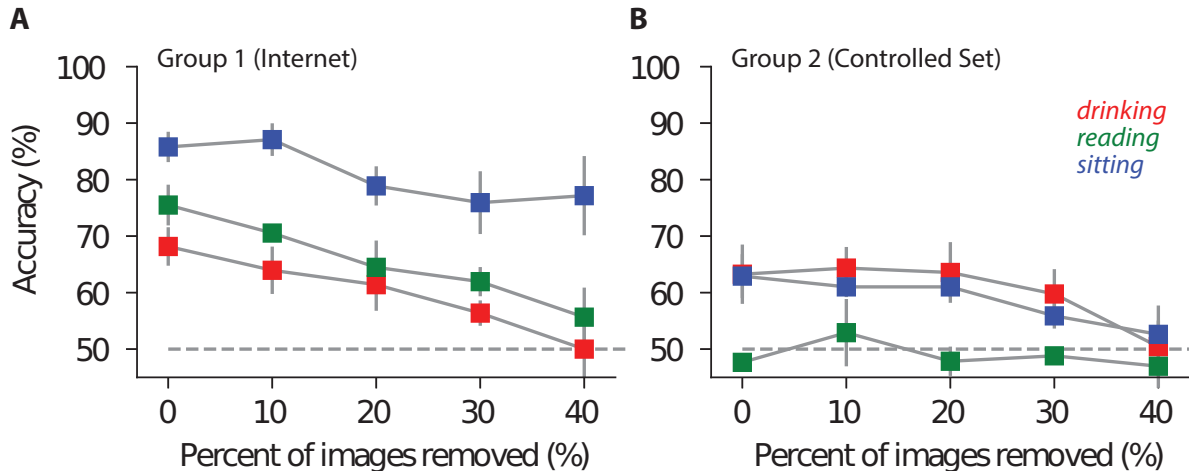
A Group 1 (Internet)

B Group 2 (Controlled Set)

drinking
reading
sitting

Figure 2. **Images downloaded from the internet carry large biases**. Accuracy on the three datasets (red=*drinking*, green=*reading*, blue=*sitting*) as a function of the percentage of images removed for images from Group 1 (**A**, Internet) or Group 2 (**B**, Controlled Set). Accuracy refers to classification results on test data using an SVM classifier on the fc7 activations of a fine-tuned AlexNet (Section 3.3). Error bars = standard deviation. Horizontal dashed line = chance level.

3.3. Removing biases

As noted in the Introduction, spurious correlations between images and labels can render tasks easy to solve. To systematically avoid such biases, we implemented a pruning procedure by ensuring that the images could not be easily classified by "simple" deep learning algorithms. This was done by applying 100 cross-validation iterations (80%/20%) of a fine-tuned AlexNet [20, 26] on each dataset. The weights were pre-trained on ImageNet [26]. A 2-unit fully-connected layer was added on top of the fc7 layer. Classification was performed by a softmax function using cross-entropy for the cost function. Weights were updated over 3 epochs, via Stochastic Gradient Descent (SGD) with momentum 0.9, L2 regularization with $\lambda = 10^{-4}$, and learning rate 10^{-4}.

After fine-tuning, an SVM was applied to the fc7 layer of fine-tuned AlexNet activations to classify the images. Images were ranked from easiest (correctly classified in most of the 100 iterations) to hardest (correctly classified only in 50% of the iterations). We progressively removed images from the dataset according to their rank and re-applied the same procedure on the reduced datasets. **Figure 2** shows the resulting drop in accuracy, as a function of the percentage of images removed.

Images from Group 1 (Internet) were easily classified (Figure 2A): accuracy was $68.2 \pm 3.4\%$ (*drinking*), $75.7 \pm 3.6\%$ (*reading*), and $85.8 \pm 2.7\%$ *sitting*), where chance is 50%, consistent with the biases inherent to Internet images. For example, the *drinking* dataset contained images of babies in the positive but not in the negative class. Other biases could be due to the surrounding environment: positive examples of *sitting* tended to correlate with indoor pictures,

whereas negative examples tended to be outdoors. After eliminating 40% of the images, *drinking* reached an accuracy of $50 \pm 5.0\%$, and *reading* reached an accuracy of $55.7 \pm 5.2\%$. In the case of *sitting*, we had to remove up to 70% of images to obtain close to chance-level accuracy.

The Group 2 dataset (our own photographs) was more difficult to classify (Figure 2B), even without any image removed: accuracy was $63.3 \pm 5.2\%$ (*drinking*), $47.7 \pm 0.8\%$ (*reading*), and $62.9 \pm 3.9\%$ (*sitting*). After eliminating 40% of the images, *drinking* reached an accuracy of $50.4 \pm 7.3\%$, and *sitting* reached an accuracy of and $52.6 \pm 2.4\%$, while *reading* dataset remained close to chance (50%).

3.4. Final dataset

After the processes in Sections 3.2 and 3.3, we obtained a final dataset for each action: 2,164 images for *drinking*, 2,524 images for *reading*, and 2,116 images for *sitting*, with 50% *yes* labels. These quantities are of the same order of magnitude as the number of images per category in the popular ImageNet dataset, where every class contains between 450 and slightly over 1,000 images. ImageNet contains many more classes (1,000 instead of the 3 x 2 classes used here). However, we note that the goal in most analyses of ImageNet is to discriminate *between* different classes. Here we are interested in detecting each action in a binary yes/no fashion, and we are not trying to discriminate one activity (e.g., *drinking*) from the others (e.g., *sitting* or *reading*). Each dataset is split into a *training* set (80%), *validation* (10%), and *test* set (10%). The persons appearing in the photographs of each set are uniquely present in that set. For example, if one person is in the *training* set, then they are not present in either the *validation* or *test* sets.
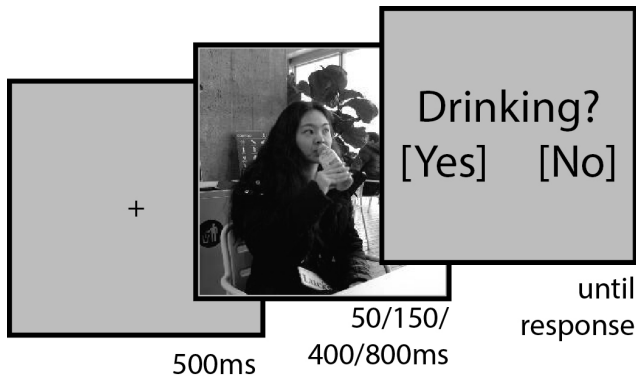
Figure 3. **Schematic description of the psychophysics task** (Section 4). Gif files were presented to mturk workers; each trial consisted of fixation (500 ms), image presentation (50, 150, 400, or 800 ms), and a forced choice yes/no question.

4. Psychophysics evaluation

Ground truth labels were obtained based on the consensus of three subjects who examined the images with no time limit (Section 3.2). To compare human versus machine performance, we conducted a separate psychophysics test with limited exposure duration of 50, 150, 400, or 800 ms in a two-alternative forced choice task implemented with psiTurk [27] (**Figure 3**). The test was delivered to a total of 54 subjects via Amazon Mechanical Turk.

The trial sequence was presented as .gif files to approximately control the duration of image presentation (Figure 3). Each trial consisted in a fixation cross (500 ms), followed by the image presented for a duration of either 50, 150, 400 or 800 ms, and finally a two-alternative forced choice question shown until the subject answered [38]. The image duration changed randomly from one presentation to the next. Despite selecting only "master mturk workers" with a rate of past accepted hits higher than 99%, online experiments often have subjects who do not fully attend or understand the task. To avoid including such cases, outlier subjects that showed a significantly lower accuracy than the population (p-value < 0.05 on one-tailed t-test) were excluded from further analyses. This threshold concerned 3 out of 18 (*drinking*), 3 out of 19 (*reading*), and 2 out of 17 (*sitting*) subjects.

The average accuracy as a function of image duration for the human subjects is shown in **Figure 4**. Even at the shortest duration (50 ms), subjects were significantly above chance in all tasks, with a performance of at least $71.8 \pm 6.1\%$ (*drinking*), up to $79.7 \pm 6.6\%$ (*sitting*). As expected, performance increased with exposure time. At the longest duration of 800 ms, performance was above 90% for all three tasks.

5. State-of-the-art models

We considered two main families of strategies to solve the task: (1) We used state-of-the-art deep convolutional neural networks pre-trained on the ImageNet dataset [31], with or without fine-tuning on the current dataset (5.1); and (2) extraction of putative action-relevant features using the Detectron algorithm [12], a state-of-the-art object-detection algorithm pre-trained on the COCO dataset [23].

5.1. Models pre-trained on ImageNet and fine-tuned on the current dataset

We considered the following deep convolutional neural networks: AlexNet [20], VGG16 [34], InceptionV3 [37], ResNetV2 [15], Inception-ResNet [36] and Xception [7] available from Keras [8]. Weights were pre-trained on ImageNet. The last classification layer, made of 1,000 units for ImageNet, was replaced by a 512x1 fc layer, followed by a 1-unit classification layer. All weights were updated via Adam optimization [19], with a learning rate of 10^{-4}, until validation accuracy stagnated. Cost was measured with binary cross-entropy and the classifier was Softmax.

We first considered the pre-trained weights followed by a classification layer. We next considered fine-tuning only the last layers. We finally considered fine-tuning the entire network with the images in the current dataset. The model yielding the highest accuracy on the validation set was applied to the test set. Results are shown in Figure 5. The top accuracy on the *drinking* dataset was $61.7 \pm 0.9\%$, obtained with the Xception network [7]. This is far below the 90.3% accuracy reached by humans on this task. Inception-ResNet [36] gave the best results for *reading* and *sitting*, with $56.7 \pm 1.8\%$ and $66.1 \pm 1.4\%$ accuracy respectively. These values are also far below the 90.7% and 94.1%, respectively, reached by humans.

We tested several additional variations in an attempt to improve performance. First, using RGB images instead of grayscale images led to similar performance, well below the accuracy obtained by humans using grayscale images (**Figure S1**). In contrast to uncontrolled datasets where color can provide strong cues (as in the skydiving versus baseball pitch example noted in the Introduction), in a more controlled dataset color does not help much. Second, accuracy was slightly improved using artificial data augmentation. Every image was horizontally flipped with probability 50%, and shifted along x or y axis by a number of pixels randomly picked in the interval [-30,30] [8]. Third, several regularization techniques were evaluated but neither L1 nor L2 normalization improved the accuracy. Finally, replacing the penultimate 512-unit fully-connected layer by 1,024 units with drop-out did not improve the accuracy either. In sum, none of the networks and variations tested here were close to human performance, even when forcing humans to use grayscale images and respond after 50 ms exposure.
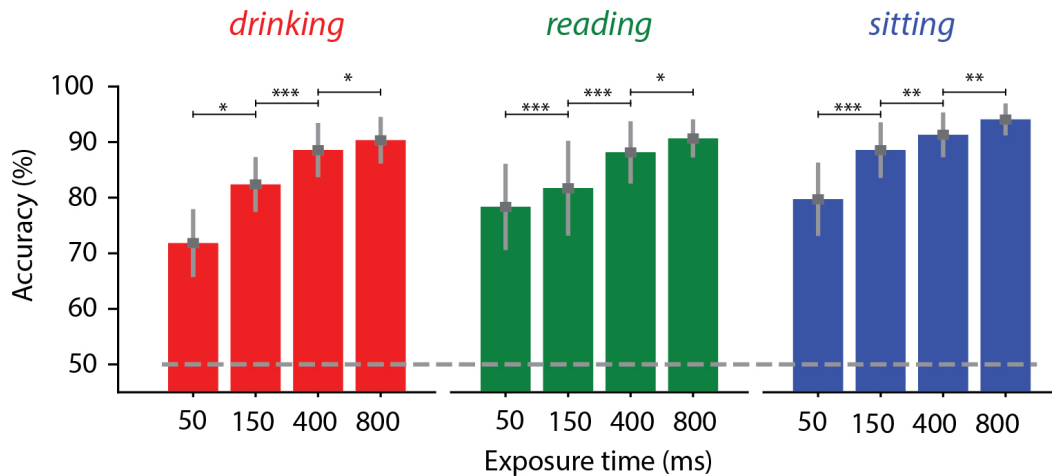
Figure 4. **Humans can rapidly detect the three actions**. Average accuracy $\pm$ SD as a function of exposure time on the three datasets in the task shown in **Figure 3**. (***) $p < 0.0005$, (**) $p < 0.05$, (*) $p < 0.1$ on one-tailed, paired t-test. Horizontal dashed line = chance level.
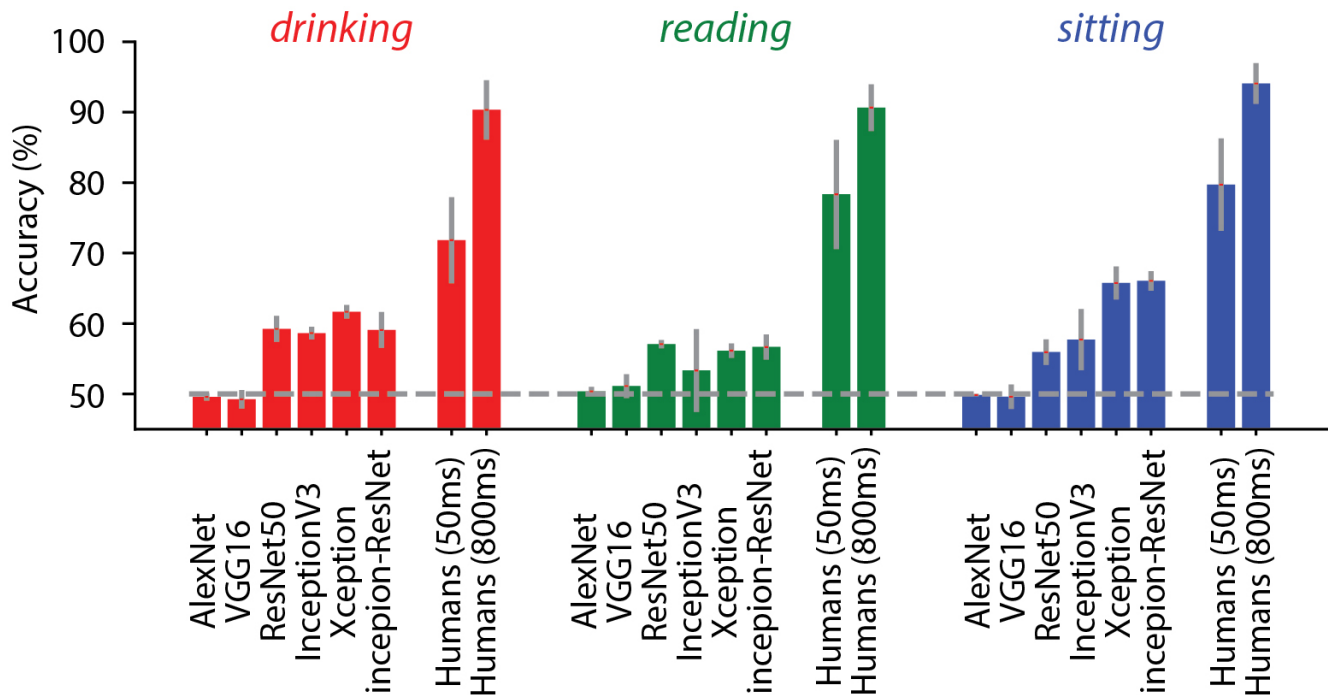

Figure 5. **Deep convolutional neural network models were far from human-level performance**. Test performance for each fine-tuned model is shown ($mean \pm SD$). The model with best accuracy on the validation set was retained to be applied on the test set, as described in section 5.1. We also reproduce here the human performance values for 50ms and 800ms exposure from **Figure 4** for comparison purposes. Human accuracy was significantly better than any of the algorithms, ($p < 0.0005$, one-tailed t-test). Horizontal dashed line = chance performance.

We visualized the salient features relevant for classification in these networks using Grad-CAM [33]. **Figure S3** shows an example visualization for the ResNet-50 network [15] with weights pre-trained on ImageNet. Even though the networks often (but not always) focused on relevant parts of the image (such as the mouth or hands for drinking), the models failed to capture the critical nuances in each image that distinguish each action. For example, reading critically depends on assessing whether the gaze is directed towards text or not.

5.2. Extraction of putative action-relevant features

Despite using a variety of state-of-the-art deep convolutional neural network architectures, with or without fine-tuning, colors, different regularizers, or data augmentation, humans outperformed all the algorithms by a large amount (**Figure 5**).

We reasoned that humans may capitalize on additional knowledge about the specific elements and interactions between elements that are involved in defining a given action. For example, *reading* depends on the presence of text (a book, a magazine, a sign), a person, and gaze directed from the person toward the text. To test this idea, we applied algorithms where we could impose the definition of each action by using computational approaches to detect the corresponding elements and their interactions.

We employed two implementations of the Detectron algorithm [12] to pursue this approach (**Figure 6**). In the first approach (Model A), we used the Detectron X-101-32x8d-FPN_s1x configuration, where 32x8d means 32 groups per convolutional layer and a bottleneck width of 8 [45], while s1x refers to the slow learning-rate schedule. This model was trained on the Keypoint Detection Task from the COCO dataset [23], comprising 150,000 person instances labelled with 17 keypoints covering their body (ankles, knees, elbows, eyes, among other points).

In the second approach (Model B), we used the Detectron X-101-64x4d-FPN_1x configuration (64 convolutional groups with a bottleneck width of 4). This model was trained for the Object Detection Task of the COCO dataset [23], consisting of 82,000 images with the objective of segmenting 81 classes of objects.

Both implementations use Mask R-CNN [14] and Feature Pyramid Network [21] for the architecture, with 101-layers ResNeXt as a backbone [45]. Both implementations obtain the highest performance in their respective tasks.

For **sitting**, only Model A was used. We extracted the bounding box, keypoints and the features of the "main" person in the picture. We defined the "main" person as the largest bounding-box whose probability of belonging to the class *person* was higher than a threshold set in the implementation. Out of the extracted data, we created two vectors: a *features* vector, made of the 12,544 features associated with the person in the picture, and a *keypoints* vector. The *keypoints* vector consisted of the x-coordinate, y-coordinate, the probability of each detected keypoint, plus the width and height of the *person* bounding-box. This resulted in a vector of 53 elements, which were normalized with respect to the bounding box coordinates. A 3 fc-layer neural network (512x1, 512x1, 2x1), trained with stochastic gradient-descent, provided the best results from the *features* vector while an SVM classifier was best for the *keypoints* vectors. The best accuracy was $76.7 \pm 2.8\%$, obtained from the *features* vectors. Grouping the two vectors together did not increase accuracy (**Figure 7**).

For **reading**, we used both models A and B. Model A was used to extract the bounding box, keypoints and the features of the main person in the picture, similarly to the *sitting* task. We used model B to extract the bounding box and features of the text material. We selected the region of interest whose probability of belonging to the classes tv, laptop, cell phone or book was higher than a certain threshold. If there were several such items in a picture, we retained the one with the largest bounding box. We combined the features from both models A and B into *features* vectors. Keypoints from models A and B were grouped into *keypoints* vectors. The same classifiers as for *sitting* were used. The best performance was reached from *keypoints* vectors with $62.8\% \pm 0.7\%$ accuracy, *features* vectors gave $56.1\% \pm 0.7\%$ accuracy.

Addressing the **drinking** task followed a similar reasoning to the *reading* task described previously. We used model A to extract the bounding box, keypoints and the features of the main person in the picture. We used model B to extract the bounding box and features of the beverage. We selected the region of interest whose probability of belonging to the classes bottle, glass, or cup was higher than a certain threshold. If there were several such items in a picture, we retained the one with the largest bounding box. We combined the features from both models A and B into *features* vectors. Keypoints from model A and B were grouped into *keypoints* vectors. The same neural network classifier as for *sitting* and *reading* was used. The best performance was reached from *features* vectors with $57.3\% \pm 1.6\%$ accuracy, while *keypoints* vectors gave $52.9\% \pm 2.6\%$ accuracy (**Figure 7**).

As discussed in Section 5.1, using RGB images instead of grayscale images led to similar accuracy, with all the models still falling below human performance levels (**Figure S2**).

6. Discussion

Can Deep Learning algorithms learn the concepts of drinking, reading, and sitting? We consider these basic activities as paradigmatic examples of daily actions that humans can recognize rapidly and seemingly effortlessly in a wide variety of different scenarios. Exciting progress in action recognition using datasets like UCF101 [35] might convey the erroneous impression that it is relatively straightforward to develop algorithms that correctly detect activities like "playing cello", "breaststroke", or "soccer juggling". However, it is important to note that algorithms can perform well above chance levels in these datasets, even simply using a linear classifier on pixel levels using just a single frame. In this work, we propose a methodology to build better controlled datasets. As a proof-of-principle, we introduce a prototype of such a dataset for the actions of *drink-*
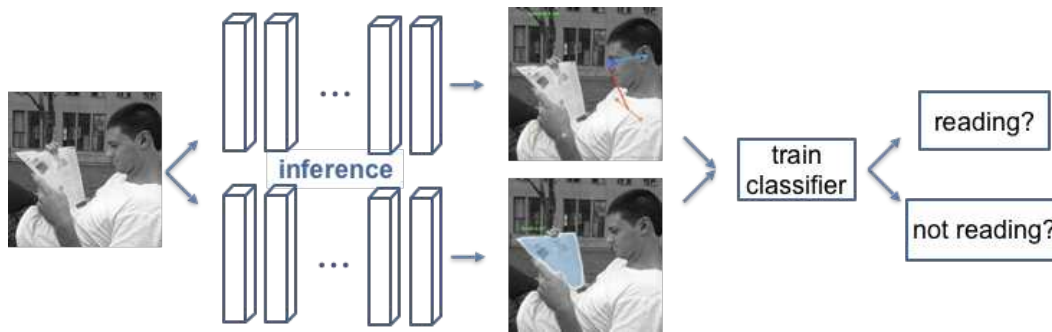
Figure 6. **Action-dependent extraction of relevant keypoints and features for *reading*.** Schematic of the implementation of Detectron [12], as described in Section 5.2. On the reading dataset, we combined two implementations of Detectron. Top: Detectron trained on the Keypoint dataset of COCO [23] allows to extract features, keypoints and bounding-box of the person in the image. Bottom: Detectron trained on the Object Detection dataset of COCO allows to extract the bounding-box and features of the reading material in the picture (see text for details).
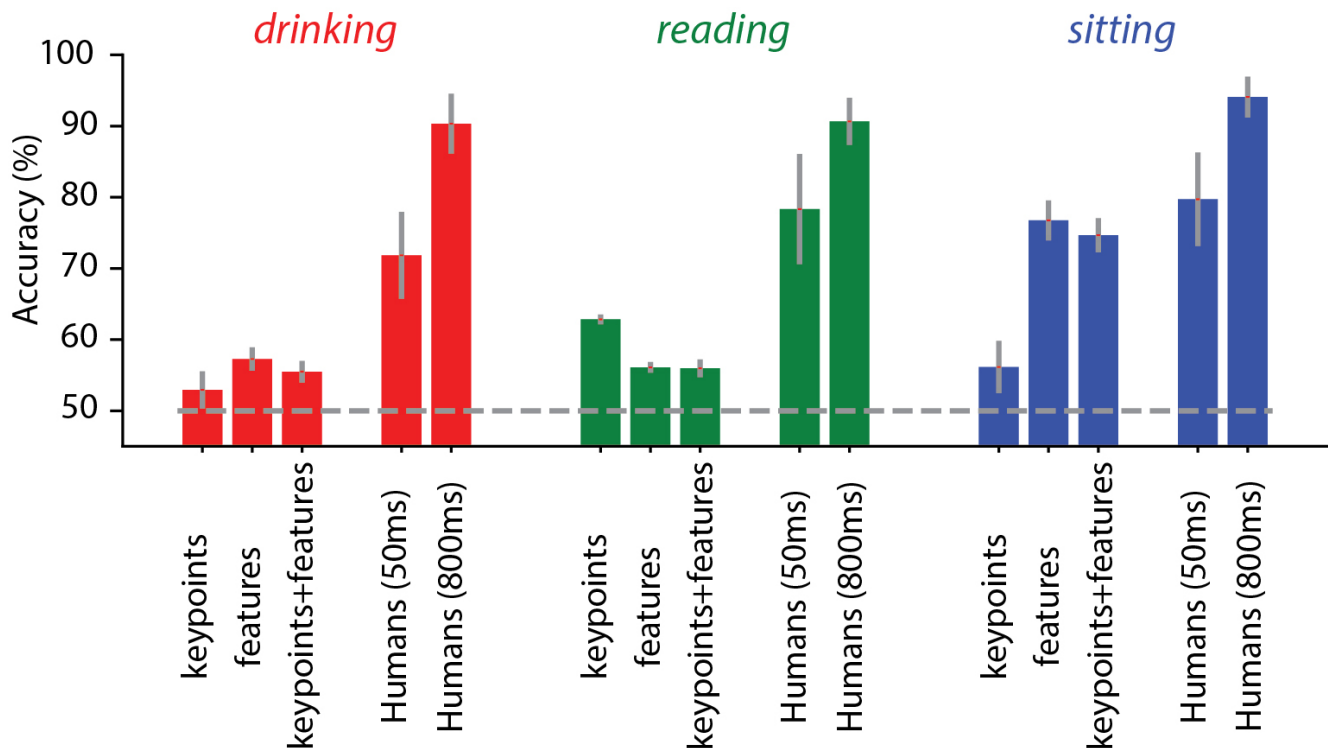

Figure 7. **Extracting action-relevant features can improve performance but all models remain well below human levels**. We extracted specific *keypoints* and *features* using the Detectron algorithm (see **Figure 6**, and text for details). The combination of action-specific keypoints and relevant object features improved performance with respect to the architectures studied in **Figure 5** for the reading and sitting datasets. Human performance with 50ms and 800ms exposure is reproduced here from **Figure 4** for comparison purposes. Horizontal line = chance performance. None of these models reached human performance levels.

ing, *reading*, and *sitting*. Using this controlled dataset, we show that the latest artificial neural networks are likely to extract some correct discriminative features as well as biased features for these behaviors and that humans outperform all of the current networks.

One approach followed by prominent datasets like Im-

ageNet [31] or UCF101 [35] is to collect example images from internet sources for a wide variety of different classes is. This approach is fruitful because it inherently represents to some extent the statistics of images in those internet sources, because there is some degree of variation captured in those images, because it enables studying multiple image

classes, and because it is empirically practical. At the same time, this approach suffers from the biases inherent to uncontrolled experiments where many confounding variables may correlate with the variables of interest [3].

Here we take a different approach whereby we consider detecting the presence or absence of specific actions. Even in this binary format, and despite our best intentions, it is difficult to download images from the internet that are devoid of biases (**Figure 2**A). For example, perhaps there are more images of people reading indoors under artificial light conditions than outdoors and therefore low-level image properties can help distinguish *reading* from *not reading* images. These biases are not always easy to infer. Regardless of the exact nature of the biases between the two classes, it is clear that images downloaded from the Internet display multiple confounding factors. In an attempt to ameliorate such biases, we took our own set of photographs under approximately standardized conditions (**Figure 1**, **Fig. S4**). This approach led to a substantial reduction in the amount of bias in the dataset (**Figure 2**B), but it was not completely bias free. Therefore, we instituted a procedure to remove images that were easy to classify.

Human subjects were still able to detect the three actions in the resulting datasets (**Figure 4**), even when exposure times were as short as 50 ms. Longer exposures led to close to ceiling performance for humans.

Computational models pre-trained on object classification datasets performed barely above chance in the three tasks (**Figure 2**B), even though the same models have been successful in the original datasets they were trained on. We re-trained state-of-the-art computational models using our datasets. Even after extensive fine tuning, data augmentation, adding color and regularizers, even the best models were well below human performance (**Figure 5**). These results should not be interpreted as a proof that no deep convolutional neural network model can reach human level performance in this dataset. On the contrary, we hope that this dataset will inspire development of better algorithms that can thrive when the number of biases is significantly reduced. An important variable in deep convolutional neural network approaches is the amount of training data. Each of our datasets contain more than 2,000 images (that is, more than 1,000 images for the *yes* and *no* classes in each case). The ImageNet dataset contains between 450 and slightly more than 1,000 images in each class. The UCF101 dataset contains on the order of 100 videos for each class. Thus, the number of images per class in our dataset is comparable or larger than the ones in prominent datasets in the field.

The total number of different tasks is very different. Here we only consider three binary tasks, whereas the typical format of object classification in ImageNet involves a single task with 1,000 classes and UCF101 involves a single task with 101 classes. Because of our binary approach, the total number of different tasks is not relevant to the results shown here. We assume that the same conclusions would apply to well-controlled datasets for other actions such as soccer juggling or not, playing cello or not, and others, but this remains to be determined. Extending our dataset creation protocol from 3 tasks to 100, or 1,000, different tasks is challenging due to the manual approach involved in taking photographs. However, recent efforts have astutely taken advantage of Amazon Mechanical Turk to collect pictures [3], an approach that could pave the way towards creating larger, yet adequately controlled, datasets.

In the interest of simplicity, here we focus on action recognition from static images as opposed to video. We were inspired to focus on static images because it is easy to thrive in current action recognition challenges by ignoring the video information. However, there is no doubt that temporal information from videos can provide a major boost to performance. Video material downloaded from the Internet suffers from similar biases to the ones discussed above for static images. Additional biases may be introduced in videos (for example, certain video classes may have more camera movement than others). It would be interesting to follow a similar approach to the one suggested here to build controlled video datasets.

The mechanisms by which human observers recognize these actions are poorly understood. It is also unclear how much class-specific training humans have with these actions. It is interesting to conjecture that many actions can be defined by an agent, an object, and a specific interaction between the two. *Drinking* involves a person (or animal), liquid, and a mechanism by which the liquid flows into the agent's mouth. Similarly, *reading* involves a person, text, and gaze directed from the person to the text. Following up on this conjecture, we provide initial steps towards defining variables of interest for action recognition using the Detectron algorithm (**Figure 6**).

When designing experiments, scientists typically devote major efforts to minimizing possible biases and confounding factors. Building less biased datasets can help challenge existing algorithms and develop better algorithms that can robustly generalize to real-world problems.

Acknowledgements

References

[1] M. Andriluka, U. Iqbal, E. Ensafutdinov, L. Pishchulin, A. Milan, J. Gall, and Schiele B. PoseTrack: A benchmark for human pose estimation and tracking. In *CVPR*, 2018. 2

[2] Mykhaylo Andriluka, Leonid Pishchulin, Peter Gehler, and Bernt Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 2

[3] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. pages 9453–9463, 2019. 8

[4] Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. OpenPose: realtime multi-person 2D pose estimation using Part Affinity Fields. In *arXiv preprint arXiv:1812.08008*, 2018. 2

[5] Joao Carreira, Pulkit Agrawal, Katerina Fragkiadaki, and Jitendra Malik. Human pose estimation with iterative error feedback, 2015. 2

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset, 2017. 1

[7] François Chollet. Xception: Deep learning with depthwise separable convolutions. *CoRR*, abs/1610.02357, 2016. 4

[8] François Chollet et al. Keras. https://keras.io, 2015. 4

[9] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks, 2016. 2

[10] Ross Girshick. Fast r-cnn, 2015. 2

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation, 2013. 2

[12] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. https://github.com/facebookresearch/detectron, 2018. 2, 4, 6, 7

[13] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions, 2017. 2

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn, 2017. 2, 6

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. *CoRR*, abs/1603.05027, 2016. 4, 5

[16] Sam Johnson and Mark Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12. 2

[17] Alex Kendall, Vijay Badrinarayanan, and Roberto Cipolla. Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding, 2015. 2

[18] Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven

Dähne. Learning how to explain neural networks: Patternnet and patternattribution, 2017. 1

[19] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. 4

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, USA, 2012. Curran Associates Inc. 1, 2, 3, 4

[21] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection, 2016. 6

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2017. 2

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. 2, 4, 6, 7

[24] Tsung-Yu Lin and Subhransu Maji. Visualizing and understanding deep texture representations, 2015. 1

[25] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[26] MathWorks. Alexnet. https://fr.mathworks.com/help/deeplearning/ref/alexnet.html. Accessed: 2019-11-12. 3

[27] Martin J.B. Markant D.B. Coenen A. Rich A.S. McDonnell, J.V. and T.M. Gureckis. *psiTurk (Version 1.02) [Software]*. New York, NY: New York University., Available from https://github.com/NYUCCL/psiTurk, 2012. 4

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

[29] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation, 2016. 2

[30] Matteo Ruggero Ronchi and Pietro Perona. Describing common human visual actions in images, 2015. 2

[31] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 1, 2, 4, 7

[32] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models, 2017. 1

[33] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016. 5

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. 4

[35] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild, 2012. 1, 2, 6, 7

[36] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning, 2016. 4

[37] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *CoRR*, abs/1512.00567, 2015. 4

[38] Lotter W Moerman C Paredes A Ortega Caro J Hardesty W Cox D Kreiman G Tang H, Schrimpf M. Recurrent computations for visual pattern completion. *PNAS*, 115:8835–8840, 2018. 4

[39] Jonathan Tompson, Ross Goroshin, Arjun Jain, Yann LeCun, and Christopher Bregler. Efficient object localization using convolutional networks, 2014. 2

[40] Alexander Toshev and Christian Szegedy. Deeppose: Human pose estimation via deep neural networks. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2014. 2

[41] Hugo Touvron, Andrea Vedaldi, Matthijs Douze, and Hervé Jégou. Fixing the train-test resolution discrepancy, 2019. 2

[42] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines, 2016. 2

[43] Bin Xiao, Haiping Wu, and Yichen Wei. Simple baselines for human pose estimation and tracking, 2018. 2

[44] Qizhe Xie, Eduard Hovy, Minh-Thang Luong, and Quoc V. Le. Self-training with noisy student improves imagenet classification, 2019. 1, 2

[45] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks, 2016. 6

[46] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks, 2013. 1