The Jester Dataset: A Large-Scale Video Dataset of Human Gestures

Joanna Materzynska  
Twenty Billion Neurons GmbH  
joanna.materzynska@twentybn.com

Guillaume Berger  
Twenty Billion Neurons GmbH  
guillaume.berger@twentybn.com

Ingo Bax  
Twenty Billion Neurons GmbH  
ingo.bax@twentybn.com

Roland Memisevic  
Twenty Billion Neurons GmbH  
roland.memisevic@twentybn.com

Abstract

Gesture recognition and its application in human-computer interfaces have been growing increasingly popular in recent years. Although many gestures can be recognized from a single image frame, to build a responsive, accurate system, that can recognize complex gestures with subtle differences between them we need large-scale real-world video datasets. In this work, we introduce the largest collection of short clips of videos of humans performing gestures in front of the camera. The dataset has been collected with the help of over 1300 different actors in their unconstrained environments. Additionally, we present an ongoing gesture recognition challenge based on our dataset and the current results. We also describe how a baseline achieving over 93% recognition accuracy can be obtained with a simple 3D convolutional neural network.

1. Introduction

Gesture recognition studies systems that can understand and categorize hand motions and use this information to control devices. Gestures are a natural and one of the oldest ways in which humans communicate. Information conveyed using gestures ranges from pointing a finger to draw attention to portraying information about space or time to signaling a need or want to one another. Hand motion is generally considered an integral part of communication just as facial expressions or language. In fact, sign language is conveyed through complex gestures and can be as versatile and complex as the spoken language. Besides facilitating the automatic processing of sign-language, gesture recognition has a wide range of applications in many industries. Gestures can be used for controlling devices in human-computer interfaces and find applications in the automotive sector, consumer electronics, public transit, gaming, home automation, and others.

Various gesture recognition technologies have been developed over the years. Wearable sensor devices, have been proposed for general recognition [8] [11] and targeted at gaming [26], and sign language [2]. They include several build-in sensors that accurately track different kinds of information, such as movement velocity, hand position, acceleration, etc. From this data, the gestures are inferred. The drawback of those approaches is the need for a device that is widely adopted or commoditized. Computer vision approaches eliminate the need for a device (besides a camera) but need large amounts of data to train systems that can generalize to unseen scenarios. Many approaches involve complex hand segmentation or joint estimation. These approaches are typically motivated by the lack of large-scale datasets that can be used to train deep neural networks. The ImageNet classification challenge was one of the first to demonstrate the usefulness of a large amount of labeled data as a way to replace complex computational pipelines by a single, end-to-end trainable model[14]. In this paper, we present experiments implicating that the vast amount of data points in our dataset attributes to the high scores obtained in our challenge.

<table>
<thead>
<tr>
<th>Dataset specification</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Total number of videos</td>
<td>148,092</td>
</tr>
<tr>
<td>Total number of frames</td>
<td>5,331,312</td>
</tr>
<tr>
<td>Number of classes</td>
<td>27</td>
</tr>
<tr>
<td>Number of actors</td>
<td>1376</td>
</tr>
<tr>
<td>Avg. duration of videos</td>
<td>3 sec</td>
</tr>
<tr>
<td>Avg. number of videos per class</td>
<td>4391</td>
</tr>
<tr>
<td>Avg. number of videos per actor</td>
<td>43</td>
</tr>
</tbody>
</table>

Table 1. Overview of the dataset statistics.

In this work, we present the first large-scale gesture recognition real-world video dataset. It is to the best of our knowledge the largest dataset of video-clips showing human gestures. It involves 148,092 short clips of videos...
Figure 1. Examples of videos from our dataset. Each image corresponds to a randomly sampled frame from a randomly sampled video. The image shows a large variance of the appearance of people, background scenes and occlusion in the videos.

of 3 seconds length, which in total account for more than 5 million frames. The video clips depict a person performing a gesture in front of the camera. In the process of data acquisition, 1,376 actors have recorded a set of 27 actions. As such, there is a significant variation in the background and appearance among actors. The gestures are complex motions that require temporal and spatial understanding, such as “Zooming In With Two Fingers” and “Zooming Out With Two Fingers” or “Pushing Hand In” and “Pushing Hand Out”. Figure 2 shows the complete list of all the gesture classes with their distributions in the dataset. We also present the models used in our ongoing video classification challenge and a simple neural network baseline model. The challenge provides an interesting survey in recent approaches for video action recognition and presents an insight into state-of-the-art architectures.

2. Related Work

Dynamic gesture recognition datasets

Existing gesture recognition datasets differ by factors such as scale, number of classes, type of annotations, sensors used and the domain of gestures. Less recent dataset, Cambridge hand gesture dataset [13], provides 900 RGB image sequences of 9 gesture classes. Sheffield Kinect Gesture (SKIG) [17] proposes a dynamic gesture dataset containing 1080 RGB-D videos collected from 6 subjects, 10 categories of gestures like (wave triangle, circle). Commonly used gesture datasets provided by the ChaLearn Gesture Challenge are ChaLearn LAP IsoGD and ConGD datasets [24], and the Multi-modal Gesture Dataset (MMGD) [5]. The gesture classes in ChaLearn LAP IsoGD and ConGD datasets are derived from 9 different domain types, from Italian sign language, activities to pantomime. Multi-modal Gesture Dataset contains 20 gesture instances.
of an Italian sign language vocabulary.

An effort that is aimed at in-car gesture recognition is described in [18], who provide driver hand gestures performed by 8 different subjects against a plain background and from a single viewpoint. There are also a variety of sign language datasets. For example, [1] present a video lexicon that should serve users to be able to lookup an entry from ASL. RWTH-BOSTON-50 [30] has been created for the experiments of isolated gesture recognition and RWTH-BOSTON-400 [4] contains 633 sequences recorded of 4 different speakers. NATOPS dataset [21] is another gesture dataset created for recognizing air signaling gestures.

Finally, the BIGHands dataset [29] is a large-scale image dataset of hand poses, it is rich in joint annotation and hand pose variation but does not directly represent gestures. Table 2 shows a comparison between the most related gesture video datasets.

Unlike previous action recognition datasets (Kinetics [3], Something-Something [7]), our dataset focuses on a small set of action categories that encompass the most commonly performed human gestures in the context of visual human-computer interfaces. With this goal in mind, the large scale of our collected dataset enables the creation of gesture recognition systems that are deployable in real-world scenarios.

Video classification We proceed to describe the models that participated in Jester Challenge in Section 4.2.

3. Large-scale gesture video dataset

In this section, we provide the dataset overview and motivation behind the chosen classes. Furthermore, we explain the acquisition procedure and crowdsourcing statistics.

3.1. Content overview

We propose a first large-scale, real-world dataset for dynamic gesture recognition. The dataset includes 148,092 video gesture clips, which is to the best of our knowledge by far the largest video-based gesture dataset to date. We propose a split into train, validation, and test set in the ratio 8:1:1. The splits are created to ensure that the videos from the same worker do not occur in both training and testing splits. Clip duration is 3 seconds. Each clip contains a gesture annotation from a set of 25 gestures used commonly...
in human-computer interfaces, including a No gesture class and a contrast class Doing other things we will describe in more detail in Section 3.2.

The videos can be downloaded at the jester-dataset website \(^1\) as videos burst into frames at 12 frames per second with a height 100px and variable width. The gestures are dynamic hand-motion patterns and in many cases cannot be distinguished from a single frame. The dataset was collected with the help of 1,376 crowd-workers. This is a much larger number of individuals than for existing datasets. The aim of the dataset was benchmarking existing gesture recognition methods as well as enabling the community to build real-time gesture recognition systems end-to-end.

**Contrast classes** Because the idea behind the dataset was to build a clip-based recognition system. There are 25 gesture classes and two classes, that should not be recognized as any particular movement. They show other actions that a user of a human-computer interface might perform without intending to communicate with the system. The No gesture category presents a video of a person sitting or standing still. The Doing other things category is a collection of various activities, such as stretching, turning head, jawing, playing with hair, etc. The crowdworkers were advised to act naturally and to perform actions other than those represented in the given gesture classes. This “catch-all” bucket for spurious and irrelevant motions makes it much easier to trade of specificity for sensitivity and thereby makes it possible to perform threshold-based recognition in a system trained solely on the clips.

**Gesture categories** Among the 25 gesture classes, there are 5 that can be described as static gestures. These could be categorized from a single frame, i.e (“Drumming Fingers”, “Thumb Up”, “Thumb Down”, ”Stop Sign”, ”Shaking Hand”). The remaining categories require distinguishing between fine-grained visual details such as “Zooming In With Two Fingers” and “Zooming In With Full Hand” or depth information, ”Rolling Hand Forward” and ”Rolling Hand Backward”.

### 3.2. Dataset Collection

For the collection of the dataset, similar to [7], we created a data collection platform that interacts with crowdsourcing services such as Amazon Mechanical Turk (AMT) to recruit crowdworkers to accept tasks and redirect them onto our platform. The task is completed and reviewed on our platform and the outcome is communicated back to the user. The outcome is either successful and results in a payment or unsuccessful, in which case a worker is allowed to re-do a task, rather than being immediately rejected.

The task for the gesture dataset is to record oneself performing all the curated gestures in front of the computer front camera. Instruction advising of visibility of hand motion, good quality of the recording, correct gestures, etc. are shown. A set of example videos is furthermore shown to clarify how the gesture is supposed to look. We found that text descriptions introduce too much ambiguity and confusion among crowdworkers and are not sufficient to convey the specifics of motions we want to capture. After receiving the textual and visual guidance on the task, a person starts recording the gesture videos. A countdown allows for getting ready until the recording of 3 seconds starts. It is possible to view the recording and perform it again if necessary. A successful submission contains approved recordings of all 27 categories. To create sufficient variance in the contrast classes, the ”Doing Other Things”-category is recorded four times, each with a different activity. The number of submissions for a single crowdworker is limited to 2. The submissions are reviewed by a human operator to ensure the correctness of the recording. Crowd workers can re-do a submission if most of the video clips were correct and only some needed correction.

### 3.3. Dataset statistics

Many existing gesture datasets, where few actors perform each gesture, often lack variability in the background. Our dataset offers a close to a real-world scenario, with a wide variety of individuals performing the gesture in the convenience of their homes. Since only the overall appearance of the gesture is given, each worker performs the gesture in the way he/she naturally would. We made sure that the hand motion is well visible, but the exact distance from the camera or the angle, left or right hand is not imposed. The total number of crowdworkers that contributed to the collection of our dataset is 1,376, the average number of videos each person recorded is 43. The task on the data platform consisted of recording all gesture classes. Only individual videos were removed if they did not meet quality standards. In Figure 3 we demonstrate examples of videos from the dataset.

### 4. Jester Challenge

To facilitate benchmarking gesture recognition models on the dataset, we published a platform where researchers can submit their test data-set predictions. On the dataset website users can anonymously submit their results to compare recognition accuracy. The ongoing challenge has gathered 59 submissions so far, 3 of which were from our team.

#### 4.1. Baseline model

For our baseline network, that currently places 34 in the challenge we propose a 3D convolutional neural network (3D-CNN). This type of network, previously described, for example, in [22, 12] uses spatio-temporal filters as the main building block. These operations provide a natural representation of spatio-temporal data. In the following, we

---

\(^1\)https://20bn.com/datasets/jester/v1/download
refer to a convolutional block as a 3d convolutional layer, followed by ReLU non-linearity and batch normalization layer. Our model consists of three 3D convolutional blocks followed by a max-pooling layer with strides operating on the spatial dimensions. We apply three more convolutional blocks and a global spatial max-pooling layer in the end. Consequently, the output of the last layer is a temporal step \( x \) feature map channels dimension vector, that we feed into a recurrent layer with LSTM cell and pass through a fully connected layer. We trained our model using SGD with a learning rate 0.001 for 100 epochs and did not implement additional data augmentation. Our model achieves 93.87% of the top 1 accuracy. The description of the networks architecture can be found in Table 3.

4.2. Methods in the challenge

Used methods in our challenge provide an interesting overview of methods used for modeling spatio-temporal activity recognition. In this section, we summarize the selected methods reported in the challenge.

Common methods

The most common approach reported in our challenge are 3D Convolutional Networks (3DCNNs) [23] [10] [27]. Ten submissions report using some variation of 3D CNN, four of them report using a 3D ResNet [9] which is a modified version of 3D CNN that uses ResNet architecture.
Each person was allowed to perform a maximum 2 submissions, however, we manually verified each set of videos and accepted a submission with few incorrect videos and deleted those from the dataset.

<table>
<thead>
<tr>
<th>layer</th>
<th>layer type</th>
<th>hyperparameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>conv3D</td>
<td>32</td>
</tr>
<tr>
<td>2</td>
<td>max pool</td>
<td>(1, 2, 2)</td>
</tr>
<tr>
<td>3</td>
<td>conv3D</td>
<td>64</td>
</tr>
<tr>
<td>4</td>
<td>max pool</td>
<td>(1, 2, 2)</td>
</tr>
<tr>
<td>5</td>
<td>conv3D</td>
<td>128</td>
</tr>
<tr>
<td>6</td>
<td>max pool</td>
<td>(1, 2, 2)</td>
</tr>
<tr>
<td>7</td>
<td>conv3D</td>
<td>256</td>
</tr>
<tr>
<td>8</td>
<td>conv3D</td>
<td>256</td>
</tr>
<tr>
<td>9</td>
<td>conv3D</td>
<td>256</td>
</tr>
<tr>
<td>10</td>
<td>global max pool</td>
<td>(1, 8, 8)</td>
</tr>
<tr>
<td>11</td>
<td>lstm</td>
<td>256</td>
</tr>
<tr>
<td>12</td>
<td>lstm</td>
<td>256</td>
</tr>
<tr>
<td>13</td>
<td>fully connected</td>
<td>256</td>
</tr>
</tbody>
</table>

Table 3. The network architecture of our baseline model. Convolution is a block of 3D convolutional layer followed by ReLU and Batch Normalization, all layers use stride 1 and filter size (3, 3, 3).

with spatio-temporal filters. The accuracy of reported 3D models ranges from 59.01% to 96.24% and the latter one is less than 1% smaller than the best performance.

**Other methods**

Two-stream networks (I3D) [3] combine the benefits of 3D CNN and two-stream networks [20]. The spatial network, used for image recognition task, is inflated to temporal dimension and now can capture motion features while second stream network, that operates on optical flow, captures recurrent information within. The submission using this approach is superior to the 3D CNN but only marginally (0.04%).

Three submissions use [32] TRN network architecture that explores temporal relations between frames. The network learns temporal relations between different number of frames and combines them at a temporal multi-scale to embed reasoning and capture both short and long term dependencies.

Building on this idea, temporal pyramid relation network [28] first extract the features with a 2D convolutional network, apply a global average pooling and use a temporal pyramid pooling before using TRN on the extracted features. We observe that it provides less than 1% improvement.

In a different way of modeling dependencies in a temporal dimension, SSNET [16] proposes a model that operates on frames and consists of a stack of dilated convolutional layers with two-dimensional filters with 14 different scales as well as a scale selection scheme that selects a subset of frames that best predict the action.

An alternative way of modeling spatio-temporal features, introduced in [15] proposes hierarchical modeling of appearance and time-window motion features. The network encodes motion and appearance from the next consecutive frame in a motion filter, merging the information from the neighboring frame and repeats this process iteratively until the information is aggregated from all hierarchical parts.

Lee et al. [15] proposes using motion features computed by optical flow. In the proposed work, the network is split into many random segments of the video. Computed in an offline manner, optical flow is then appended on a channel dimension for frames in each segment into a motion fused frames. Then from each such segment, features are extracted using ResNet. The features of each segment obtained in this manner are then concatenated and passed through another fully connected layer.

Temporal segment networks [25] also divides video in segments. Here the segments are equally long and non-overlapping. From each segment, a snipped is selected (a single frame in the paper) and an additional RGB difference and optical flow to represent motion. That spatial and temporal information is passed through separate networks in a two-stream manner and produces two outputs. The spatial and temporal scores from each segment are aggregated separately and concatenated to produce a classification score.

<table>
<thead>
<tr>
<th>Number of videos per class</th>
<th>Accuracy [top1 %]</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>62.4</td>
</tr>
<tr>
<td>200</td>
<td>71.6</td>
</tr>
<tr>
<td>500</td>
<td>77.7</td>
</tr>
<tr>
<td>1000</td>
<td>85.5</td>
</tr>
<tr>
<td>2000</td>
<td>88.3</td>
</tr>
<tr>
<td>3000</td>
<td>89.5</td>
</tr>
<tr>
<td>4391 (on average)</td>
<td>93.87</td>
</tr>
</tbody>
</table>

Table 4. Results of the experiment testing the effect of the size of the dataset on the testing accuracy. In all experiments, we used our baseline model described in section 4.1.
### Table 5. Selected entries from our leaderboard table. Time column gives a number of days between opening the leaderboard and submission.

4.3. Discussion

Analyzing the current results of our proposed challenge provides an overview of many recently proposed video classification approaches. Interestingly, 41 submissions out of 59 achieve above 90% accuracy on our dataset. One of the goals of this dataset was to provide a large enough amount of training data that gesture recognition systems could be trained that would work robustly in real-world scenarios. The strong performance of many of these methods on our test set indicates that we have accomplished this goal. However, to test the validity of this claim, we trained our baseline model using a variety of reduced training set sizes. In our experiments, we limit the number of videos per class to be: 100, 200, 500, 1000, 2000, and 3000. We observe a drastic change in network accuracy. The baseline network trained on the entire dataset achieves 93.87% of accuracy, the original split provided contains on average 4,391 videos per class, by reducing the average number by 64% to 3,000 videos per class, we observe a decrease in performance of 4.37%. Given only 100 examples per class, i.e a training set containing 2,700 videos, our baseline model achieves 62.4%, 31.47% less than when using the original data split. Table 4 provides the results of the experiments that show the
influence of data on accuracy. Given these results, we confirm that the high scores on our challenge are facilitated by a large amount of training data in our dataset. The huge variety present in our training dataset (1,376 persons, many different backgrounds) allows a variety of different methods to all adequately generalize to novel people and scenes. This is why many different methods that report results on our challenge achieve very similar performance.

5. Conclusion

We present a new large scale gesture recognition dataset. Our dataset is the largest video dataset for gesture recognition, with the most variability across actors performing the gestures. The dataset can be used to build human-computer interfaces. We also present an ongoing challenge for the classification task on our dataset. The submission platform allows users to test and compare their models across the latest state-of-the-art video recognition systems. We suggest a 3D CNN baseline model and show that the vast amount of data offered to the computer vision community significantly impacts the performance of the network, which may explain the high accuracy scores on the leaderboard.

References


