

50 Salads dataset

For an online multimedia version of this file please visit [50 Salads online](#).

Activity recognition research has shifted focus from distinguishing full-body motion patterns to recognizing complex interactions of multiple entities. Manipulative gestures - characterized by interactions between hands, tools, and manipulable objects - frequently occur in food preparation, manufacturing, and assembly tasks, and have a variety of applications including situational support, automated supervision, and skill assessment. With the aim to stimulate research on recognizing manipulative gestures we introduce the *50 Salads* dataset. It captures 25 people preparing 2 mixed salads each and contains over 4h of annotated accelerometer and RGB-D video data. Including detailed annotations, multiple sensor types, and two sequences per participant, the *50 Salads* dataset may be used for research in areas such as activity recognition, activity spotting, sequence analysis, progress tracking, sensor fusion, transfer learning, and user-adaptation.

The dataset includes

- RGB video data 640x480 pixels at 30 Hz
- Depth maps 640x480 pixels at 30 Hz
- 3-axis accelerometer data at 50 Hz of devices attached to a knife, a mixing spoon, a small spoon, a peeler, a glass, an oil bottle, and a pepper dispenser.
- Synchronization parameters for temporal alignment of video and accelerometer data
- Annotations as temporal intervals of pre- core- and post-phases of activities corresponding to steps in a recipe

An example video illustrating all available data on one sequence of preparing a salad can be found on [YouTube](#)

License

The *50 Salads* dataset is licensed under a [Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License](#). In addition, reference must be made to the following publication whenever research making use of this dataset is reported in any academic publication or research report:

Sebastian Stein and Stephen J. McKenna

[Combining Embedded Accelerometers with Computer Vision for Recognizing Food Preparation Activities](#)

The 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013),

Zurich, Switzerland, 2013. [\[.bib\]](#) [\[.pdf\]](#)

RGB Videos

Sample Files: [\[video\]](#), [\[timestamps\]](#), Download: [\[all\]](#)

Videos are encoded with DivX 5.0 compression in an .avi container. There is one timestamp file corresponding to each video. The *i*-th line in the timestamp file specifies the timestamp of the *i*-th frame in the video, and states the filename of the corresponding depth map. Note that there are some RGB frames that do not have a corresponding depth map.

Depth Maps

Sample Depth Map: [\[sample\]](#), Download: [\[all\]](#)

Depth maps are in 16-bit .pgm format with little-Endian byte-order. This is uncompressed, RGB-aligned data as acquired via OpenNI from the Kinect camera.

Loading depth in C/C++

Our code for loading depth files in C++/[OpenCV](#) can be found [here](#).

Loading depth in MATLAB

The depth-data is stored in little-Endian byte-order. In order to use it in MATLAB, use

```
depthImage = swapbytes(imread('depthImage.pgm'));
```

Accelerometer Data

Sample File: [\[sample\]](#), Download: [\[all\]](#)

The accelerometer data is in comma-separated-value format and describes a single accelerometer sample per row. Each row consists of (from left to right):

1. data info field ("ACCEL")
2. timestamp
3. device ID
4. sequence number
5. x-acceleration in g
6. y-acceleration in g
7. z-acceleration in g

Synchronization Parameters

Sample File: [\[sample\]](#), Download: [\[all\]](#)

We distinguish accelerometer time and video time. The start and end times of activity annotations correspond to video time. Video timestamps and activity start and end times need to be converted to accelerometer time using parameters specified in the synchronization files in order to associate them with accelerometer data. The synchronization files contain two types of data: (i) anchor points and offsets for converting video time to corresponding accelerometer time, and (ii) mappings from accelerometer IDs to kitchen objects.

Converting video time to accelerometer time

The first two lines of a synchronization file specify

```
[a_1], [o_1]  
[a_2], [o_2]
```

Video time is mapped to accelerometer time by linear interpolation between these anchor points using the following formula:

$$t_a = \begin{cases} t_v/1000 - o_1, & \text{if } t_v \leq a_1 \\ t_v/1000 - o_2, & \text{if } t_v \geq a_2 \\ t_v/1000 - (1 - \lambda)*o_1 - \lambda*o_2, & \text{where } \lambda = (t_v/1000 - a_1)/(a_2 - a_1) \end{cases}$$

We process accelerometer time as time of day:

```
t_a = milliseconds + 1000*(seconds + 60*(minutes + 60*(hours)));
```

Associating accelerometer IDs with kitchen objects

The synchronization file also contains a list of maps from accelerometer IDs to kitchen objects, which is separated by an empty line from anchor points and offsets. Each line in this list has the format

```
[ID],[object name]
```

Activity Annotations

Sample File: [\[sample\]](#), Download: [\[all\]](#)

Annotation files specify one annotation per line. Each annotation is in the following format:

```
[timestamp start] [timestamp end] [activity]
```

An activity corresponds to either of two levels of granularity, and each low-level activity is divided into pre-, core-, and post-phase.

High-level activity

cut_and_mix_ingredients

prepare_dressing

serve_salad

Low-level activity

peel_cucumber_{prep,core,post}

cut_cucumber_{prep,core,post}

place_cucumber_into_bowl_{prep,core,post}

cut_tomato_{prep,core,post}

place_tomato_into_bowl_{prep,core,post}

cut_cheese_{prep,core,post}

place_cheese_into_bowl_{prep,core,post}

cut_lettuce_{prep,core,post}

place_lettuce_into_bowl_{prep,core,post}

mix_ingredients_{prep,core,post}

add_oil_{prep,core,post}

add_vinegar_{prep,core,post}

add_salt_{prep,core,post}

add_pepper_{prep,core,post}

mix_dressing_{prep,core,post}

serve_salad_onto_plate_{prep,core,post}

add_dressing_{prep,core,post}

Related Publications

Publications by the dataset authors

- Sebastian Stein and Stephen J. McKenna
[Recognising Complex Activities with Histograms of Relative Tracklets](#)
Computer Vision and Image Understanding, Elsevier, 2016.
- Sebastian Stein and Stephen J. McKenna
[Combining Embedded Accelerometers with Computer Vision for Recognizing Food Preparation Activities](#)

- The 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2013), Zurich, Switzerland, 2013. [[.bib](#)] [[.pdf](#)]
- S. Stein and S. J. McKenna
[User-adaptive models for recognizing food preparation activities](#)
ACM International Conference on Multimedia (ACMMM 2013),
5th Workshop on Multimedia for Cooking and Eating Activities (CEA 2013),
Barcelona, Spain, October 21. 2013. [[.bib](#)] [[.pdf](#)]

Selected publications by other authors

We would appreciate if you dropped us an [email](#) to inform us of any publication using this dataset, so we can point to your publication on this website.

- C. Lea and R. Vidal and G. D. Hager
[Learning convolutional action primitives for fine-grained action recognition](#),
IEEE International Conference on Robotics and Automation (ICRA 2016)
- H. Kuehne, J. Gall and T. Serre
[An end-to-end generative framework for video segmentation and recognition](#),
IEEE Winter Conference on Applications of Computer Vision (WACV 2016)
- Z. Qin, K. Ren, T. Yu and J. Weng
[DPcode: Privacy-Preserving Frequent Visual Patterns Publication on Cloud](#),
IEEE Transactions on Multimedia (Volume: 18, Issue: 5, May 2016)
- Y. Yang, A. Guha, C. Fernmueller and Y. Aloimonos
[Manipulation Action Tree Bank: A Knowledge Resource for Humanoids](#),
IEEE-RAS International Conference on Humanoid Robots, Humanoids. 2014.

Acknowledgements

This research receives funding from the RCUK Digital Economy Research Hub EP/G066019/1 and [SIDE: Social Inclusion through the Digital Economy](#).

Contact

For comments, suggestions or feedback, or if you experience any problems with this website or the dataset, please contact [Stephen McKenna](#).