
epic-hand-object-detections

Release 0.0.1

EPIC-Kitchens

Mar 15, 2021

CONTENTS:

1	API Reference	1
1.1	epic_kitchens	1
2	Installation	7
3	Usage	9
4	Indices and tables	11
	Python Module Index	13
	Index	15

API REFERENCE

This page contains auto-generated API reference documentation¹.

1.1 epic_kitchens

1.1.1 Subpackages

`epic_kitchens.hoa`

Submodules

`epic_kitchens.hoa.io`

Functions for loading and saving detections to/from files.

Module Contents

Functions

`load_detections`(path: Union[(str, Path)]) → List[FrameDetections]
Load detections from file.

`save_detections`(detections: List[FrameDetections], path: Union[(str, Path)]) → None
Save detections to file.

`epic_kitchens.hoa.io.load_detections` (path: Union[str, Path]) → List[FrameDetections]
Load detections from file.

Parameters `path` – Path to detections pickle. This should contain a pickled list of serialized protobuf descriptions of detections

Returns Deserialized detections contained in pickle.

`epic_kitchens.hoa.io.save_detections` (detections: List[FrameDetections], path: Union[str, Path]) → None
Save detections to file.

Parameters

¹ Created with `sphinx-autoapi`

- **detections** – A list of detections. These should be ordered by frame.
- **path** – Path to write serialized detections to. Non-existent folders in the path are created.

`epic_kitchens.hoa.types`

The core set of types that represent hand-object detections

Module Contents

Classes

<i>HandSide</i>	Generic enumeration.
<i>HandState</i>	An enum describing the different states a hand can be in:
<i>FloatVector</i>	A floating-point 2D vector representation
<i>BBox</i>	
<i>HandDetection</i>	Dataclass representing a hand detection, consisting of a bounding box,
<i>ObjectDetection</i>	Dataclass representing an object detection, consisting of a bounding box and a
<i>FrameDetections</i>	Dataclass representing hand-object detections for a frame of a video

class `epic_kitchens.hoa.types.HandSide`

Bases: `enum.Enum`

Generic enumeration.

Derive from this class to define new enumerations.

LEFT = 0

RIGHT = 1

__repr__ (*self*)
Return repr(*self*).

__str__ (*self*)
Return str(*self*).

__dir__ (*self*)
Default dir() implementation.

__format__ (*self*, *format_spec*)
Default object formatter.

__hash__ (*self*)
Return hash(*self*).

__reduce_ex__ (*self*, *proto*)
Helper for pickle.

name (*self*)
The name of the Enum member.

value (*self*)

The value of the Enum member.

class epic_kitchens.hoa.types.**HandState**

Bases: `enum.Enum`

An enum describing the different states a hand can be in: - No contact: The hand isn't touching anything - Self contact: The hand is touching itself - Another person: The hand is touching another person - Portable object: The hand is in contact with a portable object - Stationary object: The hand is in contact with an immovable/stationary object

NO_CONTACT = 0

SELF_CONTACT = 1

ANOTHER_PERSON = 2

PORTABLE_OBJECT = 3

STATIONARY_OBJECT = 4

__repr__ (*self*)

Return repr(self).

__str__ (*self*)

Return str(self).

__dir__ (*self*)

Default dir() implementation.

__format__ (*self*, *format_spec*)

Default object formatter.

__hash__ (*self*)

Return hash(self).

__reduce_ex__ (*self*, *proto*)

Helper for pickle.

name (*self*)

The name of the Enum member.

value (*self*)

The value of the Enum member.

class epic_kitchens.hoa.types.**FloatVector**

A floating-point 2D vector representation

x :`np.float32`

y :`np.float32`

to_protobuf (*self*) → `epic_kitchens.hoa.types_pb2.FloatVector`

static from_protobuf (*vector*: `epic_kitchens.hoa.types_pb2.FloatVector`) → `epic_kitchens.hoa.types.FloatVector`

__add__ (*self*, *other*: `epic_kitchens.hoa.types.FloatVector`) → `epic_kitchens.hoa.types.FloatVector`

__mul__ (*self*, *scaler*: `float`) → `epic_kitchens.hoa.types.FloatVector`

__iter__ (*self*) → `Iterator[float]`

property coord (*self*) → `Tuple[float, float]`

Return coordinates as a tuple

scale (*self*, *width_factor*: *float* = 1, *height_factor*: *float* = 1) → None
Scale x component by *width_factor* and y component by *height_factor*

class epic_kitchens.hoa.types.BBox

left :float

top :float

right :float

bottom :float

to_protobuf (*self*) → epic_kitchens.hoa.types_pb2.BBox

static from_protobuf (*bbox*: epic_kitchens.hoa.types_pb2.BBox) →
epic_kitchens.hoa.types.BBox

property center (*self*) → Tuple[float, float]

property center_int (*self*) → Tuple[int, int]
Get center position as a tuple of integers (rounded)

scale (*self*, *width_factor*: *float* = 1, *height_factor*: *float* = 1) → None

center_scale (*self*, *width_factor*: *float* = 1, *height_factor*: *float* = 1) → None

property coords (*self*) → Tuple[Tuple[float, float], Tuple[float, float]]

property coords_int (*self*) → Tuple[Tuple[int, int], Tuple[int, int]]

property width (*self*) → float

property height (*self*) → float

property top_left (*self*) → Tuple[float, float]

property bottom_right (*self*) → Tuple[float, float]

property top_left_int (*self*) → Tuple[int, int]

property bottom_right_int (*self*) → Tuple[int, int]

class epic_kitchens.hoa.types.HandDetection

Dataclass representing a hand detection, consisting of a bounding box, a score (representing the model's confidence this is a hand), the predicted state of the hand, whether this is a left/right hand, and a predicted offset to the interacted object if the hand is interacting.

bbox :BBox

score :np.float32

state :HandState

side :HandSide

object_offset :FloatVector

to_protobuf (*self*) → epic_kitchens.hoa.types_pb2.HandDetection

static from_protobuf (*detection*: epic_kitchens.hoa.types_pb2.HandDetection) →
epic_kitchens.hoa.types.HandDetection

scale (*self*, *width_factor*: *float* = 1, *height_factor*: *float* = 1) → None

center_scale (*self*, *width_factor*: *float* = 1, *height_factor*: *float* = 1) → None


```

class epic_kitchens.hoa.types.ObjectDetection
    Dataclass representing an object detection, consisting of a bounding box and a score (the model's confidence
    this is an object)

    bbox :BBox

    score :np.float32

    to_protobuf (self) → epic_kitchens.hoa.types_pb2.ObjectDetection

    static from_protobuf (detection:          epic_kitchens.hoa.types_pb2.ObjectDetection)      →
                           epic_kitchens.hoa.types.ObjectDetection

    scale (self, width_factor: float = 1, height_factor: float = 1) → None

    center_scale (self, width_factor: float = 1, height_factor: float = 1) → None

class epic_kitchens.hoa.types.FrameDetections
    Dataclass representing hand-object detections for a frame of a video

    video_id :str

    frame_number :int

    objects :List[ObjectDetection]

    hands :List[HandDetection]

    to_protobuf (self) → epic_kitchens.hoa.types_pb2.Detections

    static from_protobuf (detections:          epic_kitchens.hoa.types_pb2.Detections)      →
                           epic_kitchens.hoa.types.FrameDetections

    static from_protobuf_str (pb_str: bytes) → epic_kitchens.hoa.types.FrameDetections

    get_hand_object_interactions (self, object_threshold: float = 0, hand_threshold: float = 0)
                                   → Dict[int, int]
        Match the hands to objects based on the hand offset vector that the model uses to predict the location of
        the interacted object.

        Parameters

        • object_threshold – Object score threshold above which to consider objects for
          matching

        • hand_threshold – Hand score threshold above which to consider hands for matching.

        Returns A dictionary mapping hand detections to objects by indices

    scale (self, width_factor: float = 1, height_factor: float = 1) → None
        Scale the coordinates of all the hands/objects. x components are multiplied by the width_factor and
        y components by the height_factor

    center_scale (self, width_factor: float = 1, height_factor: float = 1) → None
        Scale all the hands/objects about their center points.

```

epic_kitchens.hoa.visualisation

Tools for visualising hand-object detections

Module Contents

Classes

DetectionRenderer

A class to render hand-object annotations onto the corresponding image

```
class epic_kitchens.hoa.visualisation.DetectionRenderer (hand_threshold: float = 0.8,  
                                                         object_threshold: float = 0.01,  
                                                         only_interacted_objects: bool = True,  
                                                         font_size=20, border=4,  
                                                         text_padding=4)
```

A class to render hand-object annotations onto the corresponding image

```
render_detections (self, frame: PIL.Image.Image, detections:  
                    epic_kitchens.hoa.types.FrameDetections)  $\rightarrow$  PIL.Image.Image
```

Parameters

- **frame** – Frame to annotate with hand and object detections
- **detections** – Detections for the current frame

Returns A copy of `frame` annotated with the detections from `detections`.

INSTALLATION

```
$ pip install git+https://github.com/epic-kitchens/epic-kitchens-100-hand-object-  
↪bboxes.git
```


Visualise the hand-object detections like so:

```
from pathlib import Path
import PIL.Image
from epic_kitchens.hoa.io import load_detections
from epic_kitchens.hoa.visualisation import DetectionRenderer

class LazyFrameLoader:
    def __init__(self, path: Union[Path, str], frame_template: str = 'frame_{:010d}.
↪jpg'):
        self.path = Path(path)
        self.frame_template = frame_template

    def __getitem__(self, idx: int) -> PIL.Image.Image:
        return PIL.Image.open(str(self.path / self.frame_template.format(idx + 1)))

detections = load_detections('detections/P01_101.pkl')
frames = LazyFrameLoader('frames/P01_101')
renderer = DetectionRenderer(hand_threshold=0.5, object_threshold=0.5)

frame_idx = 100
renderer.render_detections(frames[frame_idx], detections[frame_idx])
```

An Jupyter notebook example is included that demonstrates how to detections and visualise them.

INDICES AND TABLES

- `genindex`
- `search`

PYTHON MODULE INDEX

e

- `epic_kitchens`, [1](#)
- `epic_kitchens.hoa`, [1](#)
- `epic_kitchens.hoa.io`, [1](#)
- `epic_kitchens.hoa.types`, [2](#)
- `epic_kitchens.hoa.visualisation`, [6](#)

Symbols

`__add__()` (*epic_kitchens.hoa.types.FloatVector method*), 3
`__dir__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__dir__()` (*epic_kitchens.hoa.types.HandState method*), 3
`__format__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__format__()` (*epic_kitchens.hoa.types.HandState method*), 3
`__hash__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__hash__()` (*epic_kitchens.hoa.types.HandState method*), 3
`__iter__()` (*epic_kitchens.hoa.types.FloatVector method*), 3
`__mul__()` (*epic_kitchens.hoa.types.FloatVector method*), 3
`__reduce_ex__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__reduce_ex__()` (*epic_kitchens.hoa.types.HandState method*), 3
`__repr__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__repr__()` (*epic_kitchens.hoa.types.HandState method*), 3
`__str__()` (*epic_kitchens.hoa.types.HandSide method*), 2
`__str__()` (*epic_kitchens.hoa.types.HandState method*), 3

A

`ANOTHER_PERSON` (*epic_kitchens.hoa.types.HandState attribute*), 3

B

`BBox` (*class in epic_kitchens.hoa.types*), 4
`bbox` (*epic_kitchens.hoa.types.HandDetection attribute*), 4
`bbox` (*epic_kitchens.hoa.types.ObjectDetection attribute*), 5

`bottom` (*epic_kitchens.hoa.types.BBox attribute*), 4
`bottom_right()` (*epic_kitchens.hoa.types.BBox property*), 4
`bottom_right_int()` (*epic_kitchens.hoa.types.BBox property*), 4

C

`center()` (*epic_kitchens.hoa.types.BBox property*), 4
`center_int()` (*epic_kitchens.hoa.types.BBox property*), 4
`center_scale()` (*epic_kitchens.hoa.types.BBox method*), 4
`center_scale()` (*epic_kitchens.hoa.types.FrameDetections method*), 5
`center_scale()` (*epic_kitchens.hoa.types.HandDetection method*), 4
`center_scale()` (*epic_kitchens.hoa.types.ObjectDetection method*), 5
`coord()` (*epic_kitchens.hoa.types.FloatVector property*), 3
`coords()` (*epic_kitchens.hoa.types.BBox property*), 4
`coords_int()` (*epic_kitchens.hoa.types.BBox property*), 4

D

`DetectionRenderer` (*class in epic_kitchens.hoa.visualisation*), 6

E

`epic_kitchens`
 module, 1
`epic_kitchens.hoa`
 module, 1
`epic_kitchens.hoa.io`
 module, 1
`epic_kitchens.hoa.types`
 module, 2
`epic_kitchens.hoa.visualisation`
 module, 6

F

`FloatVector` (*class in epic_kitchens.hoa.types*), 3

frame_number (*epic_kitchens.hoa.types.FrameDetection* *ObjectDetection* (class in *epic_kitchens.hoa.types*),
 attribute), 5 4
 FrameDetections (class in *epic_kitchens.hoa.types*), 5 objects (*epic_kitchens.hoa.types.FrameDetections* attribute), 5
 from_protobuf () (*epic_kitchens.hoa.types.BBox* static method), 4 P
 from_protobuf () (*epic_kitchens.hoa.types.FloatVector* *PORTABLE_OBJECT* (*epic_kitchens.hoa.types.HandState* static method), 3 attribute), 3
 from_protobuf () (*epic_kitchens.hoa.types.FrameDetections* static method), 5 R
 from_protobuf () (*epic_kitchens.hoa.types.HandDetection* static method), 4 render_detections ()
 from_protobuf () (*epic_kitchens.hoa.types.ObjectDetection* static method), 5 method), 6
 from_protobuf_str ()
 (*epic_kitchens.hoa.types.FrameDetections* static method), 5
 G
 get_hand_object_interactions ()
 (*epic_kitchens.hoa.types.FrameDetections* method), 5
 H
 HandDetection (class in *epic_kitchens.hoa.types*), 4
 hands (*epic_kitchens.hoa.types.FrameDetections* attribute), 5
 HandSide (class in *epic_kitchens.hoa.types*), 2
 HandState (class in *epic_kitchens.hoa.types*), 3
 height () (*epic_kitchens.hoa.types.BBox* property), 4
 L
 left (*epic_kitchens.hoa.types.BBox* attribute), 4
 LEFT (*epic_kitchens.hoa.types.HandSide* attribute), 2
 load_detections () (in module
epic_kitchens.hoa.io), 1
 M
 module
 epic_kitchens, 1
 epic_kitchens.hoa, 1
 epic_kitchens.hoa.io, 1
 epic_kitchens.hoa.types, 2
 epic_kitchens.hoa.visualisation, 6
 N
 name () (*epic_kitchens.hoa.types.HandSide* method), 2
 name () (*epic_kitchens.hoa.types.HandState* method), 3
 NO_CONTACT (*epic_kitchens.hoa.types.HandState* attribute), 3
 O
 object_offset (*epic_kitchens.hoa.types.HandDetection* attribute), 4
 S
 save_detections () (in module
epic_kitchens.hoa.io), 1
 scale () (*epic_kitchens.hoa.types.BBox* method), 4
 scale () (*epic_kitchens.hoa.types.FloatVector* method), 3
 scale () (*epic_kitchens.hoa.types.FrameDetections* method), 5
 scale () (*epic_kitchens.hoa.types.HandDetection* method), 4
 scale () (*epic_kitchens.hoa.types.ObjectDetection* method), 5
 score (*epic_kitchens.hoa.types.HandDetection* attribute), 4
 score (*epic_kitchens.hoa.types.ObjectDetection* attribute), 5
 SELF_CONTACT (*epic_kitchens.hoa.types.HandState* attribute), 3
 side (*epic_kitchens.hoa.types.HandDetection* attribute), 4
 state (*epic_kitchens.hoa.types.HandDetection* attribute), 4
 STATIONARY_OBJECT
 (*epic_kitchens.hoa.types.HandState* attribute), 3
 T
 to_protobuf () (*epic_kitchens.hoa.types.BBox* method), 4
 to_protobuf () (*epic_kitchens.hoa.types.FloatVector* method), 3
 to_protobuf () (*epic_kitchens.hoa.types.FrameDetections* method), 5
 to_protobuf () (*epic_kitchens.hoa.types.HandDetection* method), 4
 to_protobuf () (*epic_kitchens.hoa.types.ObjectDetection* method), 5
 top (*epic_kitchens.hoa.types.BBox* attribute), 4

`top_left()` (*epic_kitchens.hoa.types.BBox* property),
4
`top_left_int()` (*epic_kitchens.hoa.types.BBox*
property), 4

V

`value()` (*epic_kitchens.hoa.types.HandSide* method), 2
`value()` (*epic_kitchens.hoa.types.HandState* method),
3
`video_id` (*epic_kitchens.hoa.types.FrameDetections*
attribute), 5

W

`width()` (*epic_kitchens.hoa.types.BBox* property), 4

X

`x` (*epic_kitchens.hoa.types.FloatVector* attribute), 3

Y

`y` (*epic_kitchens.hoa.types.FloatVector* attribute), 3