

Project Note AP06-2019-01

OpenPose in the Public DGS Corpus



Authors: [Marc Schulder](#), [Thomas Hanke](#)

DOI: [10.25592/uhhfdm.842](https://doi.org/10.25592/uhhfdm.842)

Releases:

v1.0 (9 Dec 2019): Original version for corpus release 2.

v2.0 (11 May 2020): Update for corpus release 3.

Abstract

This project note gives an overview of how pose information was created for the Public DGS Corpus with the use of OpenPose. Pose information is machine-readable data that describes where people are located in an image, providing the coordinates for various points of each body, such as joints, eyes or ears. The data we generate consists of body, face and hand models for informants in every camera perspective of all published transcripts.

Several postprocessing steps were applied to the data before publication. These include a) the correction of errors, such as the false positive recognition of bodies, and recognition of one actual person as two distinct bodies, b) ensuring the consistent order of people across frames in multi-person perspectives, c) the removal of pose information for the moderator, and d) the anonymisation of utterances containing sensitive information.

The resulting data is stored in one JSON file per transcript. Each file contains the pose information of the three published camera perspectives. The data format is designed to collect the default single-frame outputs of OpenPose in a single file and to provide additional relevant metadata for each camera perspective.

Contents

1 Introduction	3
2 OpenPose	4
2.1 Keypoints	4
2.2 Data Format	4
3 OpenPose on the Public DGS Corpus	7
3.1 Processing the corpus	8
3.2 Video Total: The Side-View Perspective	9
3.3 Correction of Recognition Errors	10
3.4 Anonymisation	11
3.5 Data Format	12
3.5.1 Missing Frames & People	14
3.6 Conversion to Single Frame Format	14
4 Licence	15
References	16

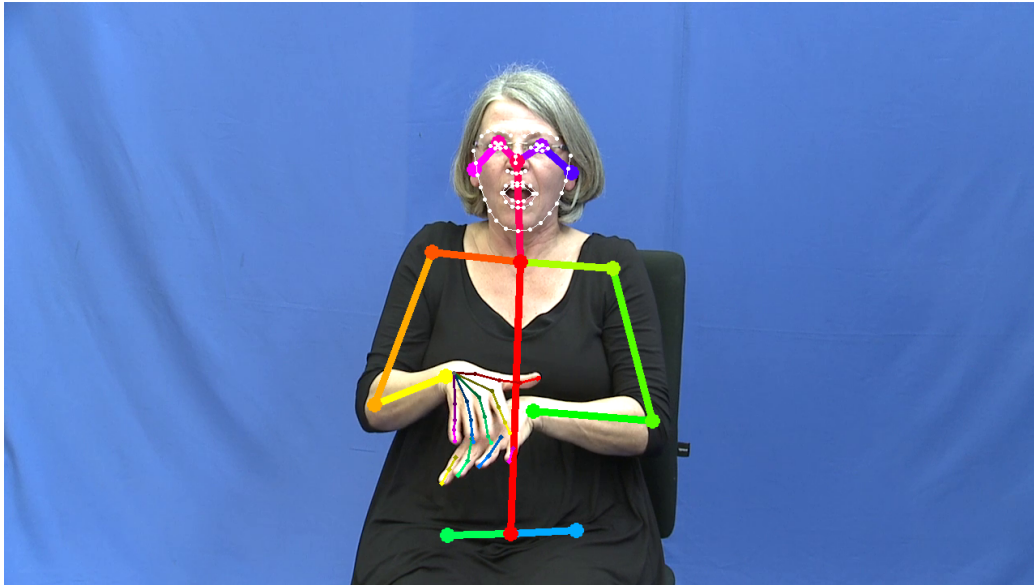


Figure 1: Visual representation of OpenPose pose information, computed for a video from the Public DGS Corpus. Sets of keypoints are generated for the body, face and each hand. Lines between points are added to the visual representation to indicate the logical connection between individual keypoints.

1 Introduction

The Public DGS Corpus (Konrad et al., 2020) provides pose recognition data for all its video recordings. This data offers users explicit machine-readable information on the location of various body parts, such as hands, shoulders, nose, ears, individual finger joints and others. Pose information was generated automatically using OpenPose (Cao et al., 2018; Simon et al., 2017). A visual representation of the computed information can be seen in [Figure 1](#).

In [Section 2](#) we provide an introduction to OpenPose, describing how it represents body poses in an image ([Section 2.1](#)) and how its data is formatted ([Section 2.2](#)). [Section 3](#) describes how OpenPose is applied to the data of the Public DGS Corpus. We discuss processing the corpus ([Section 3.1](#)), applying anonymisation ([Section 3.4](#)), the format in which we provide the information ([Section 3.5](#)) and how it can be converted to the original OpenPose format ([Section 3.6](#)). [Section 4](#) contains a brief reminder of the license conditions of the Public DGS Corpus.

2 OpenPose

OpenPose is a pose estimation tool that determines the location of human bodies in an image, down to individual joints (shoulder, wrist, knee, etc.) and other points of interest like the nose and eyes. By applying OpenPose to frames of a video, movement estimation can be provided for individual body parts.

In [Section 2.1](#) we describe how the position of body parts is described through keypoints. [Section 2.2](#) discusses the data format in which OpenPose outputs its predictions.

2.1 Keypoints

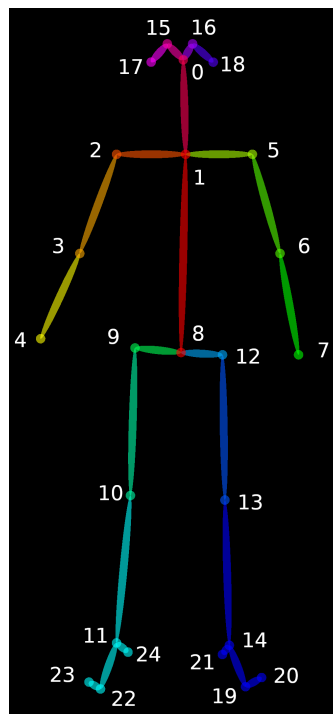
Each human body is represented by a number of **keypoints**. Each keypoint represents a skeletal joint or an otherwise relevant part of the body. For example, in the body model (shown in [Figure 2a](#)) the keypoint with index 0 represents the location of the nose, index 2 the right shoulder joint and index 8 the center of the hip.

The main OpenPose model by [Cao et al. \(2018\)](#) computes a general skeleton of the body, which identifies the joints in the shoulder, arms, hip and legs, as well as a few keypoints in the face and feet (see [Figure 2a](#)). [Simon et al. \(2017\)](#) add additional models to compute detailed hand and face models (see [Figures 2b](#) and [2c](#) respectively).

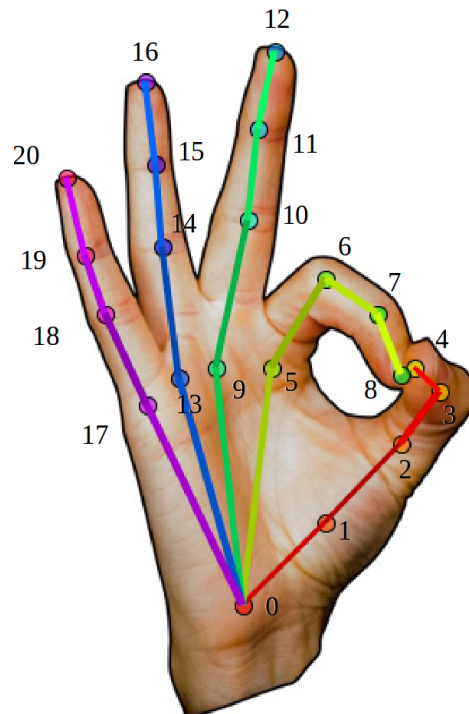
In the case of the body model, two models are available: the default BODY_25, which provides 25 keypoints, and the COCO model, which reduces complexity of the hip and legs and provides 18 keypoints. We use the BODY_25 model, which is recommended for most applications.

2.2 Data Format

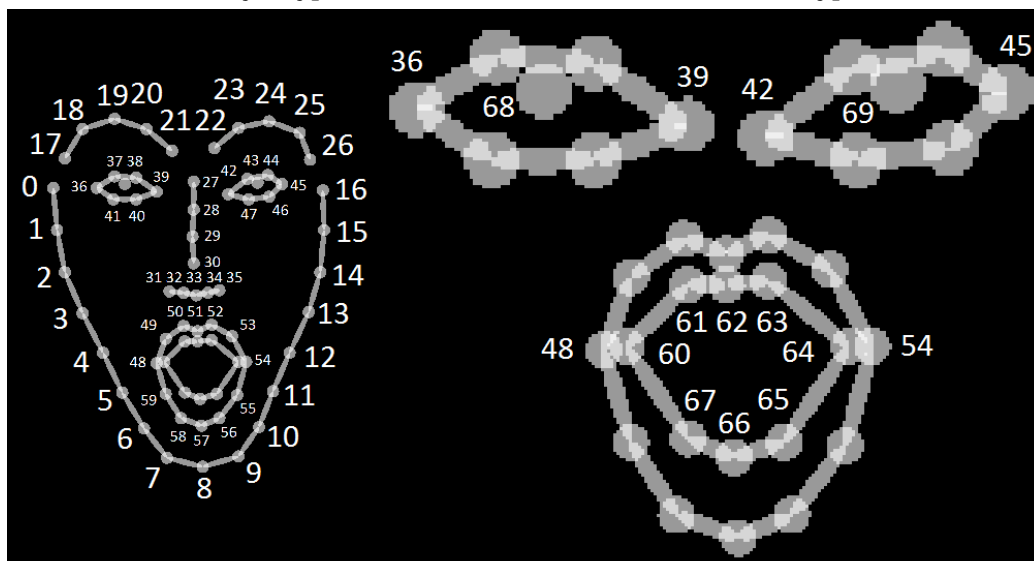
For an example of the output format of OpenPose, see [Listing 1](#). OpenPose creates an output file for each processed frame. Each file reports the OpenPose version used to generate the data and a list of people that were detected in the frame. Each person is represented as a mapping to eight lists, which represent 2D and 3D keypoints of the pose (i. e. the body), face, left hand and right hand. In our case only the 2D estimation was performed, so the four 3D keypoint lists are always empty.



(a) Body keypoints



(b) Hand keypoints



(c) Face keypoints

Image source: <https://github.com/CMU-Perceptual-Computing-Lab/openpose>

Figure 2: Skeleton maps of the keypoints determined by OpenPose. The number provided for each keypoint indicates its location in the output file (see Listing 1).

Project Note AP06-2019-01

Listing 1: An abridged example of an OpenPose output file for a single frame containing one person. In this case, only two-dimensional recognition was performed. Each line of numbers represents a single keypoint. A keypoint consists of an x-coordinate, a y-coordinate and a confidence value. Keypoints that were not visible in the frame are represented as "0,0,0". For reasons of space, this example omits many keypoints and shows "... " instead.

```
1 { "version":1.4,
2   "people":[
3     { "pose_keypoints_2d":[
4       662.642,184.701,0.844587,
5       666.649,290.495,0.756731,
6       545.137,304.218,0.569373,
7       ...
8       0,0,0,
9       0,0,0],
10    "face_keypoints_2d":[
11      591.583,194.741,0.447692,
12      ...
13      670.916,155.074,0.909006],
14    "hand_left_keypoints_2d":[
15      760.857,472.859,0.385735,
16      ...
17      701.18,509.802,0.631602],
18    "hand_right_keypoints_2d":[
19      541.926,255.422,0.562895,
20      ...
21      596.183,166.286,0.624597],
22    "pose_keypoints_3d":[],
23    "face_keypoints_3d":[],
24    "hand_left_keypoints_3d":[],
25    "hand_right_keypoints_3d":[]
26  }
27 ]
28 }
```

A keypoint is represented as an x-coordinate, a y-coordinate and a confidence value. Accordingly, each keypoint list represents its n keypoints as a series of $3*n$ decimal numbers where the first three numbers represent the first keypoint, the fourth till sixth the second keypoint etc. The x- and y-coordinates specify pixel positions of the keypoint in the image.¹ Digits after the decimal point signify that the coordinate lies between two pixels. As is customary for image coordinates, the x-coordinate goes from left to right and the y-coordinate from top to bottom. The confidence value represents the classifier's confidence that the keypoint is positioned correctly. It is given as a decimal value between 0 (no confidence) and 1 (complete confidence).² If a keypoint could not be detected (e.g. because the body part in question is not visible in the image) its three components are all set to 0.

For an example, observe the frame output for a video with a resolution of 1280×720 pixels in [Listing 1](#). It was computed with OpenPose version 1.4 and detected a single person. The body gives the location of the nose (keypoint index 0, see [Figure 2a](#)) as being at pixel position 662.642×184.701 . This means the nose is almost at the horizontal center and 25% below the top of the image. OpenPose has a high confidence of 84.4587% that this position is correct. The final keypoints of the body model (right toe and right ankle, indices 23 and 24), on the other hand, are marked as missing as they are not visible in the picture.

3 OpenPose on the Public DGS Corpus

Pose information was computed for all transcripts of the Public DGS Corpus. The information for the front-facing recordings of the informants (*Video A* and *Video B*) was published in corpus release 2 ([Konrad et al., 2019](#)). Information for the side-view that shows both informants and the moderator (*Video Total*) was added in release 3 ([Konrad et al., 2020](#)). We provide keypoints for the body, face and both hands of all informants. Pose information for moderators (visible in the side-view

¹OpenPose also offers normalised keypoint coordinates, but we choose to let the user decide whether such post-processing steps are desirable for their application.

²In some cases the confidence value may in fact be slightly below 0 or above 1. While the ground-truth training data provides confidence values in the range from 0 to 1, the recognition algorithm is not explicitly constrained to this range. Source: <https://github.com/CMU-Perceptual-Computing-Lab/openpose/issues/615>

Project Note AP06-2019-01

Transcript	Age Group	Format	Topics	iLex File	ELAN File	Film A	Film B	Film Total	SRT File	Film AB	OpenPose
dgskorpus_ber_01	18-30	Experience of Deaf Individuals	Communication: Communication with Hearing People Deaf Culture								

Figure 3: File selection on ling.meine-dgs.de. OpenPose data for each transcript is offered in the final column.

recordings) is not released. This is in line with the decision to only translate but not transcribe moderator utterances.

The resulting pose information is available on ling.meine-dgs.de as part of the files offered for each transcript (see [Figure 3](#)). For each transcript there exists a single file containing all its OpenPose data. The file is contained in a compiled gzip archive to reduce its download size.

The remainder of this section provides information specific to the Public DGS Corpus release of OpenPose data. [Section 3.1](#) discusses how OpenPose was run to process the corpus data. [Section 3.2](#) provides information specific to the side-view perspective of *Video Total* recordings. [Section 3.4](#) details how sections that are anonymised in the public release were handled. [Section 3.5](#) introduces the format of the data wrapper which bundles all frames of a transcript. In [Section 3.6](#) we discuss how the original OpenPose frame data can be extracted from the data wrapper.

3.1 Processing the corpus

OpenPose was run on the original full-size recordings of the DGS-Korpus project. For more technical details on how OpenPose was run on a high performance cluster to process the corpus data, see [Hanke \(2019\)](#).

The resulting data was reviewed for dropped frames and erroneous detections of too many bodies. For example, as the front-facing cameras record a single person, detections of multiple bodies were almost always in error, except for a few cases where another participant leaned into the image. These false positives were subsequently removed from the data.

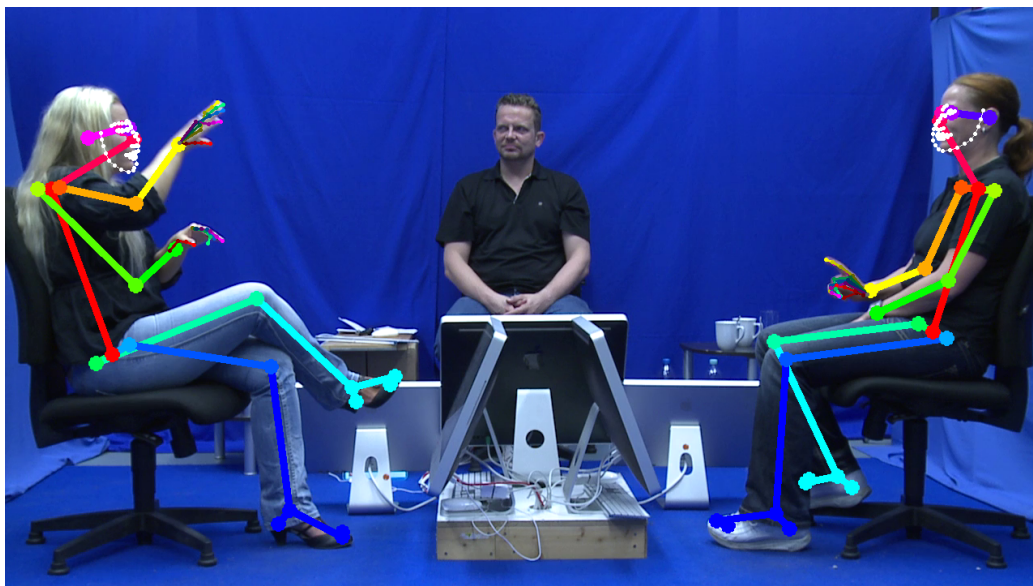


Figure 4: Side-view perspective offered by Video Total recordings, overlaid with published pose information. Informant B can be seen on the left, informant A on the right. The moderator can be seen sitting in the background in the centre of the image.

3.2 Video Total: The Side-View Perspective

Corpus release 3 (Konrad et al., 2020) introduces pose information for *Video Total*, the side-view perspective of the dialogue, which shows both participants facing each other and the moderator sitting between them in the background. This perspective complements the pose information for *Video A* and *Video B*, the front-facing recordings of informants A and B, which was published as part of corpus release 2 (Konrad et al., 2019).

Figure 4 shows an example of a *Video Total* recording with overlaid pose information. In these recordings informant B is always situated on the left and informant A on the right. The moderator is seated in the background in the centre of the image. Pose information is provided for informants, but not for the moderator, as the moderator is not considered a language informant for the corpus and is therefore not subject to linguistic analysis.

In OpenPose multiple people in a frame are represented as a series

of entries under the top-level key "people". The order in which OpenPose lists people is not always consistent across frames. For the corpus data we automatically verified and corrected the order (see [Section 3.3](#)) to ensure that people are always listed from left to right, i. e. the first entry represents informant B (sitting on the left) and the second entry represents informant A (sitting on the right).³

For frames in which one of the people was erroneously not recognised by OpenPose, their keypoint models are represented as a list of zeroes (i. e. no known keypoint coordinates). This ensures that accessing a specific index in the people list always returns the same person across all frames of a transcript.

3.3 Correction of Recognition Errors

Due to the controlled environment of the corpus recordings we know how many people should be detected in each image. Where more than the expected number of people was detected, we observed one of three issues:

1. **ghost recognitions** in which additional bodies are recognised where there are none,
2. **fragmented recognitions** in which body parts of a single participant are detected as the keypoints of multiple persons, e. g. the keypoints of the left arm and head are recognised as one person, the keypoints of the torso and right arm as a second person and the waist and legs as belonging to a third person,
3. **accidental appearances** of a second person in the image, e. g. when an informant would raise their hand high enough to show up in the frontal recording of the other informant.

Ghost recognitions and accidental appearances were automatically removed. Fragmented recognitions were automatically merged back into a single person entry. The code used to correct these errors in the total perspective is made publicly available.⁴

³The exception to this are transcripts with a single participant, e. g. in the category *Jokes*. In these only the relevant informant is listed. So a joke told by participant A will list them as the first and only person, just like in their frontal perspective, rather than including an additional person entry for informant B.

⁴<https://github.com/DGS-Korpus/Public-Corpus-OpenPose-error-corrector>



Figure 5: *Anonymised utterance in the Public DGS Corpus. Both the right hand and mouth are anonymised to obfuscate both sign and mouthing. In other cases, only one or the other or the entire body may be obfuscated.*

3.4 Anonymisation

The Public DGS Corpus has been anonymised to protect personal information, such as the names of people (Bleicken et al., 2016). Relevant elements of the annotation were marked as containing sensitive information. Based on these, glosses, mouthings, translations, videos and pose information were anonymised. In the video data this is achieved by drawing black rectangles over the parts of the video that require anonymisation. Figure 5 shows an example of this. As pose information was not yet available when video anonymisation was originally performed, the locations of the black rectangles was determined manually.

Pose recognition for the corpus was performed on the original recordings, so anonymisation is applied as a post-processing step. We differentiate between two cases:

1. Mouth anonymisation.
2. Complete anonymisation.

id	The ID of the transcript that the recording belongs to.
camera	The camera perspective that the recording represents. a1 for the front-facing recording of informant A, b1 for the front-facing recording of informant B and c for the side-view recording of both informants.
width	The pixel width of the recording.
height	The pixel height of the recording.
frames	A key-value mapping from frame index (starting at 0) to the complete OpenPose output of that frame (see Section 2.2). The mapping lists all frames of the associated video file in which OpenPose detected at least one person.

Table 1: *The top-level fields of each recording entry of the wrapper format. The fields id, camera, width and height are recording metadata applicable to each frame. For an example of how these fields are filled, see [Listing 2](#).*

In the case of mouth anonymisation we remove all face model keypoints of the anonymised person in the given frames. For complete anonymisation, we remove the keypoints of all four models, effectively removing all pose information of the person for the affected frames.

Complete anonymisation also covers cases in which a single hand was anonymised. This is because the area obfuscated is usually larger, covering the arm and possibly shoulder and face. In such cases we choose to err on the side of caution and remove all pose information, rather than reveal keypoints that annotators intended to anonymise.

If a model is anonymised, its list of keypoints is replaced with an empty list, just like the 3D models in [Listing 1](#). This allows the user to differentiate it from naturally invisible keypoints, which are shown as triples of zeroes, rather than as missing data. This approach also preserves the number of people entries for a frame, which is especially important for the side-view perspective (see [Section 3.2](#)).

3.5 Data Format

By default, OpenPose generates one file per frame (see [Section 2.2](#)). As we cover many hours of recordings, this would result in a huge number of very small files. This is not advisable, as file systems can be

Project Note AP06-2019-01

Listing 2: *An abridged example of pose information for the corpus transcript with the ID 1413451-11105600-11163240, containing three recordings. Frame information is omitted here (marked as "..."), but an example can be seen in [Listing 1](#).*

```
1 [ {
2   "id": "1413451-11105600-11163240",
3   "camera": "a1",
4   "width": 1280,
5   "height": 720,
6   "frames": {
7     "0": { ... },
8     "1": { ... },
9     ...
10    "16840": { ... }
11  }
12 }, {
13   "id": "1413451-11105600-11163240",
14   "camera": "b1",
15   "width": 1280,
16   "height": 720
17   "frames": {
18     "0": { ... },
19     ...
20    "16840": { ... }
21  }
22 }, {
23   "id": "1413451-11105600-11163240",
24   "camera": "c",
25   "width": 1280,
26   "height": 720,
27   "frames": {
28     "0": { ... },
29     ...
30    "16840": { ... }
31  }
32 }
33 ]
```

markedly slower at handling many small files than they are at handling small numbers of large files. Instead, we create a wrapper format that collects all frames for all recordings of a transcript. One file contains pose information for *Video A*, *Video B* and *Video Total* (see [Figure 3](#)). For each recording we provide relevant metadata in addition to the OpenPose output. [Table 1](#) provides a description of the top-level fields of a recording. [Listing 2](#) shows an example of the resulting wrapper.

3.5.1 Missing Frames & People

When OpenPose does not detect any person in a given frame, it does not create an output file for that frame. Accordingly, we do not create a key-value pair for it. Users should therefore explicitly refer to the frame keys when iterating over a series of frames, rather than assuming an unbroken sequence of frames.

In cases where a frame key-value pair is given, but the person entry or entries have only empty keypoint lists, this indicates that OpenPose detected people, but the information was anonymised (see [Section 3.4](#)).

When a frame key-value pair is given, but the keypoint lists of a person are completely filled with zeroes, this indicates that the person was not recognised by OpenPose, but their entry was added in post-processing to ensure consistent person indices across frames (see [Section 3.2](#)). In some cases you may encounter frames that only list such unrecognised people. This is caused by OpenPose recognising the moderator or ghosts, but not recognising the participants. This can be treated as functionally equivalent to a missing frame key-value pair.

3.6 Conversion to Single Frame Format

To convert the data from the wrapper format described in [Section 3.5](#) back to the standard OpenPose output ([Section 2.2](#)), we provide a Python converter script.⁵ It requires at least Python 2.7 or Python 3 to run.

The script takes one or several wrapper files as input and for each input file creates a directory of frame files. The name of the output directory matches the name of the input file, unless an alternative directory is provided.

⁵<https://github.com/DGS-Korpus/Public-Corpus-OpenPose-frame-extractor>

Project Note AP06-2019-01

To retain the additional metadata of the wrapper, the created frame files use the following naming pattern:

```
id_camera.widthxheight.frame_frame.keypoints.json
```

Words in bold represent variables that are filled with the five meta-data values from [Section 3.5](#). In the case of **frame**, the frame index is given as a 13-digit value with prepended zeroes, matching the output of OpenPose.

For example, applying the conversion script to the pose file for transcript *1413451-11105600-11163240* shown in [Listing 2](#) would result in a directory containing 50,520 files, representing the three sets of 16,840 frames from the *Video A*, *Video B* and *Video Total* recordings for this transcript. The very first file in the directory would be the first frame of *Video A* and have the following filename (split into two lines for reasons of space):

```
1413451-11105600-11163240_a1.1289x720.  
frame_0000000000000000.keypoints.json
```

The filename for the second frame would be identical to the first, except it would say `frame_0000000000000001` instead of `frame_0000000000000000`.

4 Licence

The OpenPose data of the Public DGS Corpus falls under the same licence and usage restrictions as the remaining parts of the Public DGS Corpus.⁶ It may only be downloaded and used for linguistic research. Should you publish research based on the material, please cite the corresponding publications.⁷

⁶http://ling.meine-dgs.de/license_en.html

⁷<http://dgs-korpus.de/index.php/publications.html>

References

- Bleicken, Julian, Thomas Hanke, Uta Salden, and Sven Wagner (2016). “Using a Language Technology Infrastructure for German in order to Anonymize German Sign Language Corpus Data”. In: *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*. Portorož, Slovenia, pp. 3303–3306. URL: <https://www.aclweb.org/anthology/L16-1526>.
- Cao, Zhe, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh (2018). *OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields*. Preprint. arXiv: [1812.08008 \[cs.CV\]](https://arxiv.org/abs/1812.08008).
- Hanke, Thomas (2019). *Processing DGS-Korpus Data with OpenPose on the Hamburg High Performance Cluster*. Project Note AP04-2018-01. Hamburg, Germany: DGS-Korpus project, IDGS, Hamburg University. DOI: [10.25592/uhhfdm.834](https://doi.org/10.25592/uhhfdm.834).
- Konrad, Reiner, Thomas Hanke, Gabriele Langer, Dolly Blanck, Julian Bleicken, Ilona Hofmann, Olga Jeziorski, Lutz König, Susanne König, Rie Nishio, Anja Regen, Uta Salden, Sven Wagner, and Satu Worseck (2019). *MY DGS – Annotated. Public Corpus of German Sign Language, 2nd Release*. Dataset. DGS-Korpus project, IDGS, Hamburg University. DOI: [10.25592/dgs.corpus-2.0](https://doi.org/10.25592/dgs.corpus-2.0).
- Konrad, Reiner, Thomas Hanke, Gabriele Langer, Dolly Blanck, Julian Bleicken, Ilona Hofmann, Olga Jeziorski, Lutz König, Susanne König, Rie Nishio, Anja Regen, Uta Salden, Sven Wagner, Satu Worseck, and Marc Schulder (2020). *MY DGS – Annotated. Public Corpus of German Sign Language, 3rd Release*. Dataset. DGS-Korpus project, IDGS, Hamburg University. DOI: [10.25592/dgs.corpus-3.0](https://doi.org/10.25592/dgs.corpus-3.0).
- Simon, Tomas, Hanbyul Joo, Iain Matthews, and Yaser Sheikh (2017). “Hand Keypoint Detection in Single Images Using Multiview Bootstrapping”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Honolulu, Hawaii, USA, pp. 4645–4653. DOI: [10.1109/CVPR.2017.494](https://doi.org/10.1109/CVPR.2017.494).