Annotation Facilities for the Reliable Analysis of Human Motion

Michael Kipp

University of Applied Sciences Augsburg
An der Hochschule 1
86161 Augsburg, Germany
michael.kipp@hs-augsburg.de

Abstract

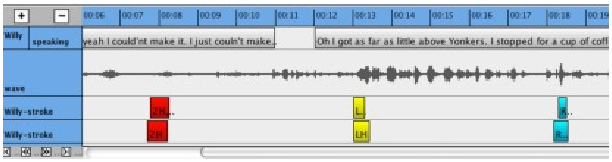
Human motion is challenging to analyze due to the many degrees of freedom of the human body. While the qualitative analysis of human motion lies at the core of many research fields, including multimodal communication, it is still hard to achieve reliable results when human coders transcribe motion with abstract categories. In this paper we tackle this problem in two respects. First, we provide facilities for qualitative and quantitative comparison of annotations. Second, we provide facilities for exploring highly precise recordings of human motion (motion capture) using a low-cost consumer device (Kinect). We present visualization and analysis methods, integrated in the existing ANVIL video annotation tool (Kipp, 2001), and provide both a precision analysis and a "cookbook" for Kinect-based motion analysis.

Keywords: Multimodal Communication, Annotation Tool, Human Motion Analysis

1. Reliability

The existing ANVIL video annotation tool is mostly used in the area of multimodal communication research, often involving the modalities of speech and various categories of body motion (most notably gesture). The tool has recently been extended by motion capture playback and reliability analysis features (Kipp, 2012b; Kipp, 2012a). The purpose of this paper is to introduce a lost-cost method to capture motion capture data using a Kinect. Moreover, since we argue that motion capture brings more objectivity to coding we also present a number of features that potentially increase the reliability of coding.

Transcribing human motion usually involves a certain amount of interpretation on the side of the human coder. It is therefore not surprising that reported reliability measures are quite low. For instance, in the area of gesture research, it has proven quite hard to consistently code *movement phases* (preparation, stroke, retraction etc.) (Kita et al., 1998): It is difficult to exactly pinpoint the time point where one phase transitions to another (segmentation problem) but also to determine the category based on objective criteria (classification problem). The reliability of coding depends, on the one hand, on how cleanly defined the coding scheme is but also, on the other hand, on information and facilities that the annotation tool provides. Facilities for analyzing coders' disagreement are important to provide feedback to human coders and to improve the definition of poorly performing categories in the coding scheme.



yeah I could'nt make it. I just couln't make...

Oh I got as far as little above Yonkers. I stopped for a cup of coff

Figure 1: Comparing annotations of track "Willy-stroke" by two different coders in time alignment.

Bottom-up vs. top-down Coding In multimodal communication research, it is common that information pieces relate to each other in terms of a subsumption or constituency relationship. Consider the encoding of syllables, words and sentences on different tiers. Syllables are subsumed by a word, words are subsumed by a sentence. In gesture research, gestural motion is often decomposed into small motion units called *phases* that constitute a larger overall motion, called a *phrase* (Kita et al., 1998). This can be operationalized in coding by defining two tracks, one for phases, one for phrases, and imposing a constituency relationship between the two tracks. With *top-down coding* we refer to the method of first defining the larger segments (phrases) and then subdividing those into smaller units (phases). *Bottom-up coding* works vice versa, i.e. first identifying the small segments (phases) and then joining them into the larger phrases. In ANVIL, bottom-up coding has been realized using the *span* relationship. We now added the *subdivision* relationship[1] which allows top-down coding. ANVIL also offers *point elements* which contain only a single time point. For a number of coding schemes this is necessary (e.g. if only interested in the onset of a signal) and providing this annotation type contributes to keeping the coding non-redundant and thus less error-prone. Providing a wide range of track relationships also facilitates interoperability with other annotation tools like ELAN or Exmaralda (Schmidt et al., 2008).

Qualitative side-by-side comparison of codings ANVIL has an integrated facility for computing reliability scores, both for single file comparison and whole corpora (Kipp, 2012b). While numerical measures of agreement like kappa are important to get an overall impression of coding reliability (Carletta, 1996), such measures are of limited help when it comes to identifying the *causes* for disagreement. For this, a qualitative analysis is necessary, directly com-

[1] We deliberately chose the same name as in the ELAN tool (Wittenburg et al., 2006) to make clear that our subdivision relationship is identical to ELAN's definition of the concept.

paring the different codings. We have implemented a flexible way of doing this by allowing to insert arbitrary tracks from other files into the current annotation file. Thus, the coder can then see the coding of two different coders in time alignment (Fig. 1). The coder can also hide and resize arbitrary tracks to enhance the visibility of the two (or more) tracks under investigation. Ideally, this should be complemented with automatic methods that highlight "hot spots", i.e. regions of strong disagreement. Note that there is always the principal distinction of segmentation disagreement (where are the boundaries of elements) and categorical disagreement. Both aspects need to be addressed.

2. Motion Analysis

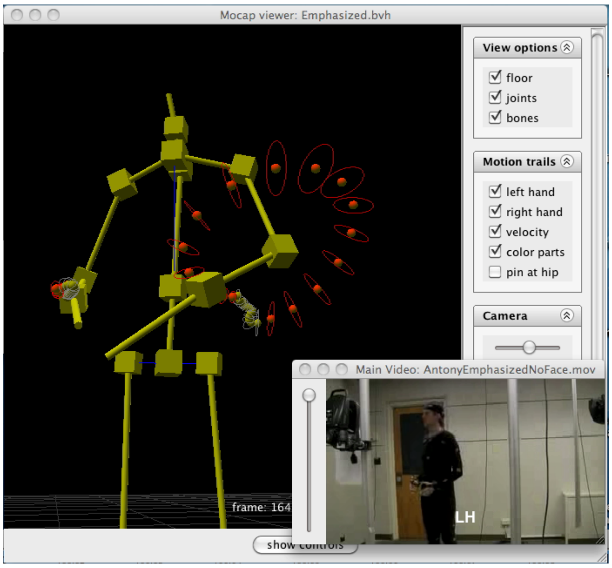
2.1. Visualization

In a previous publication we suggested to visualize the path of a hand movement as *motion trails*, color-coded in the colors of the annotation elements on the timeline (Heloir et al., 2010). Similar visualizations can be found in tools for 3D animation, e.g. in the open-source Blender[2] software where they are called *motion paths*.

Velocity visualization To become an effective tool in manual annotation, the 3D view must integrate as much information as possible without cluttering the view. Therefore, to add velocity and motion direction, we introduced *motion circles* (Fig. 2). While the motion direction is orthogonal to the circle's plane, the radius of the circle is proportional to the current speed. We found this visualization both intuitive and non-distracting.

Local frames of reference The human body is often viewed as a hierarchical articulated structure of rigid bones, connected by rotational joints. Any point in space (including e.g. the position of the hand) can be expressed either in the global "world" frame of reference or in the local frame of reference of a particular joint. This is relevant to coding because frequently the motion of the hand may be more meaningful if looked on as in the local frame of reference of the upper body[3], instead of using the global "world" frame of reference. Therefore, we introduce the notion of "pinning" the skeleton. It means that the skeleton is fixed at the hip joint (like a butterfly on a pin) so that all upper boy motion is relative to the hip joint. Fig. 3 shows a motion where the hand is actually *almost still* but because the hip moves (posture shift) the hand floats to the side which is reflected in the motion trail (left frame). If pinning is active, this motion is much reduced (right frame) as one would expect, given that the hand is still when seen relative to the body. Note that whether pinning makes sense may depend on the situation. The hip motion may deliberately be used by the performer to move the hand or the hip motion may be a "coincidence". Our tool lets motion analysts make the "pinning" decision with the switch of a button. Note also that there are multiple ways to perform *pinning* using other frames of reference (e.g. using the shoulder joints or the thorax joint).

Possible future extensions For the future, we envision to extend the visualization by derived measures. A *tangent arrow* could facilitate the detection of discontinuities. One could also highlight the key point with the highest amount of direction change, possibly in a color-coded fashion like a heat map (of course, this would be a selectable alternative view to the current timeline color-coding).


Mocap viewer: Emphasized.bvh
Main Video: AntonyEmphasizedNoFace.mov
frame: 164
LH
show controls

View options
☑ floor
☑ joints
☑ bones

Motion trails
☑ left hand
☑ right hand
☑ velocity
☑ color parts
□ pin at hip

Camera

Figure 2: Circles depict velocity direction (orthogonal to the circle's plane) and amount (proportional to its radius).

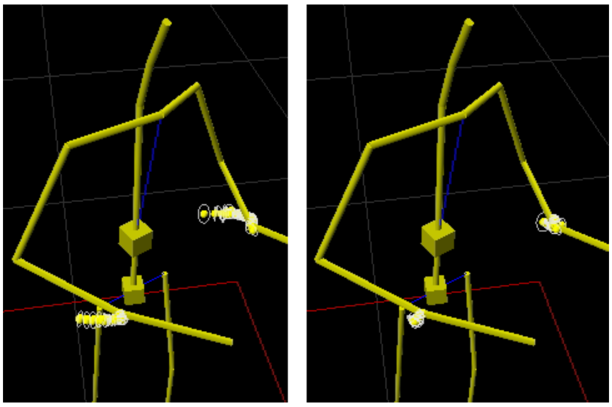


Figure 3: The same hand motion trails without pinning (left) and with pinning at the hip joing (right).

2.2. Continuous Data and Objective Measures

As mentioned before, coding reliability primarily depends on the crisp definition of categories. Motion capture for the first time allows to actually use objective numerical information to do this. On the basis of continuous data, i.e. the angular changes in the joints, which is in turn sampled with a certain frame rate (usually between 25-50 frames per sec), we are able to derive objective measures and thresholds to guide manual coding.

For instance, a "discontinuity" in the motion path of the hand can be mathematically defined. Even much simpler measure like the height of the hand (in relation to the hip) or the distance of the hand from the body have the potential of making coding much more precise and reliable. Looking at multi-party interactions, such objective measures can be

[2] www.blender.org

[3] This could be realized using e.g. the hip joint or the thorax joint.

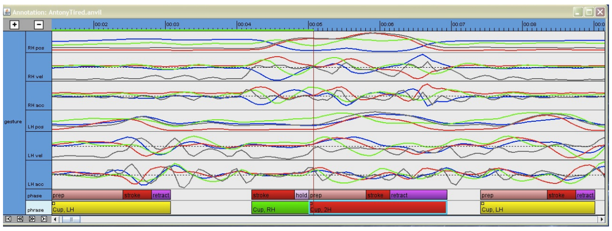


Figure 4: Motion curves (position, velocity, acceleration) for both hands.

extended to interpersonal distance, orientation and mutual gaze.

In speech processing, it is common to use the visualization of the waveform (i.e. loudness) to guide word and syllable segmentation. Not only does motion visualization allow to proceed in an analogous fashion, we can also proceed to explore correlations between two continuous signal data, speech and motion, which are at the core of multimodal communication research.

Currently, we are able to visualize various transformations of the motion path of the hands to support annotation (Fig. 4). First of all, we can show curves for hand position (x, y, z), velocity (x, y, z and overall) and acceleration (x, y, z and overall). Moreover, we have defined a range of signal filters that can be applied to these curves to experimentally explore ways of automatic coding. These filters include noise filters and thresholding filters.

For the future we envision a tool for visually composing filters and letting ANVIL automatically fill a track according to user-defined criteria (rules) and automatically conduct a quantitative comparison with a predefined Gold Standard. A first target of investigation is the automatic segmentation and classification of gesture phases.

2.3. Automatic Coding

Here, we report results on the rather simple task of detecting the handedness of a gesture (Heloir et al., 2010). To detect handedness (LH, RH, 2H) on the phrase level (i.e. for a whole gesture) we first find the corresponding *expressive phase* on the phase track. The expressive phase is either a stroke or an independent hold (Kita et al., 1998). We take the length of the path travelled by left hand L_{RH} and right hand L_{LH} respectively during this expressive phase (in meters), and normalize it by the duration d of the phase (in seconds). If the normalized difference $\frac{|L_{RH}-L_{LH}|}{d}$ is below the threshold of $0.12\frac{m}{s}$ (value was found experimentally), we label it a bihanded gesture (2H), otherwise we label it right-handed if $L_{RH} > L_{LH}$, or left-handed (LH) if $L_{RH} < L_{LH}$. On an annotated corpus of 269 phrases, we achieved 83% correct annotations with this algorithm.

3. Motion Capture

In this section we describe how to perform motion capture for ANVIL using the Kinect device. We also point out some limitations and present precision experiments for hand positions.

3.1. Kinect Recording Cookbook

For the moment we have only conducted single-person recordings. We assume that both motion capture and traditional video recordings are of interest for the annotation process. The result of any recording session are two files, one traditional video file (e.g. Quicktime or AVI) and one motion capture file in BVH format. These two media must then be synchronized in Anvil.

Note that the Kinect has a limited angle of recording so that, for a full body recording, a distance of 2-3 meters from the Kinect has to be maintained. For the recording, one needs to set up a Microsoft Kinect device and a digital camcorder (we use a HD webcam).

In terms of software, you need the following (this setup only runs under Windows, we only list free software):

- Brekel Kinect[4], including the OpenNI auto installer (installs all necessary middleware)
- Software for webcam recording (e.g. Debut Video Capture[5])
- Video conversion tool to produce an ANVIL compatible codec: VirtualDub or MPEG Streamclip

To record the BVH file do the following:

1. Start Brekel which initializes the Kinect. The screen will show camera image and depth image.
2. Go into "Psi pose", i.e. stretch out both arms to the side and lift lower arms upward such that upper arm and lower arm are perpendicular. Brekel will show when the skeleton is recognized.
3. Press "Start capture BVH".
4. Perform a synchronization movement (see below).
5. When finished, press "Stop".

The resulting BVH file is compatible with ANVIL. We will not describe how to prepare the video (see ANVIL documentation). The synchronization between BVH and video in ANVIL must be done manually in ANVIL (again, see documentation). For this synchronization it is important to have a good *synchronization movement* that is clearly visible in both video and mocap, for instance outstretching arms to the side and immediately retract. Please make sure that during recording no other people or human-like shapes (e.g. posters, paintings in background) are visible to the Kinect since this may result in ghost skeletons in the BVH file.

3.2. Limitations and Precision

Limitations Kinect recognizes the body structure using a depth image that is inferred from what the two infrared cameras in the device capture. If visual occlusion occurs, depth cannot be inferred any more. For instance, a hand held behind the back will make it impossible for the Kinect

[4] www.brekel.com

[5] www.nchsoftware.com/capture

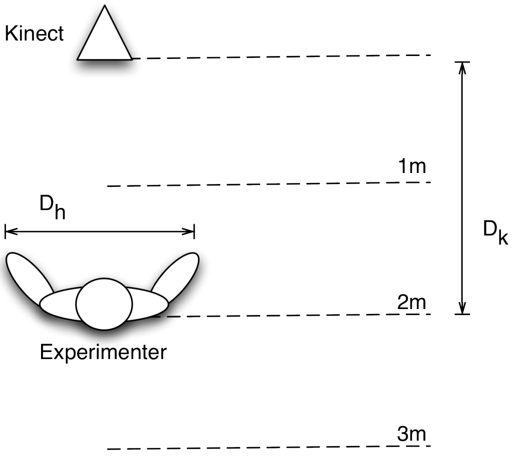


Figure 5: Setup of Kinect precision measurements.

to recognize the hand. Also, when the two hands are close together or touching, the Kinect may not be able to distinguish the two objects and will likely lose track. In our experiments we found that hands need to be held apart about 20cm to be reliably recognized. Distances of 10-20cm deliver mixed results and hands closer than 10cm usually confuse the Kinect.

Precision/Spatial resolution Since the Kinect is a low-cost consumer device, it is important to know how precisely motion is measured. One key aspect is spatial resolution: how much deviation occurs when measuring the same point in space multiple times? We conducted a series of experiments to measure this. The setup was as follows (see Fig. 5): a person stood in front of the Kinect and had markings for a fixed hand-to-hand distance we call D_h (1 meter or 1.5 meters). The person would hold the two hands apart with a distance of D_h and then moved the hands together and apart again several times. In analysis, we looked at the maximum distance between the hands for every time the person held the hands apart. The variation constitutes our tolerance measure. We tried this with different distances D_k between person and Kinect (1, 2 and 3 meters). Our results showed that there was a maximum deviation of 1-1.5 cm. In our opinion, this error is tolerable in the area of gesture research.

4. Future Work

In this section, we summarize our future efforts that we have mentioned throughout the text. For reliability analysis we introduced a side-by-side view of the same track, annotated by different coders. A future extension will highlight "hot spots" of disagreement, both in terms of segmentation and classification.

For our motion capture visualization we plan to augment the visualization by displaying tangent arrows and marking the peak velocity of a path segment.

Finally, the display of motion curves (position, velocity, acceleration) will be complemented by an array of filters (e.g. thresholding) that can be used to automatically fill a track with discrete annotation segments. This technique will be used to tackle the problem of automatic gesture phase detection.

5. Conclusions

In this paper we have presented various facilities for improving the precision and reliability of human motion coding. First, we presented reliability computations and qualitative analysis features. Second, we presented motion capture visualizations and showed how to record motion capture using the low-budget Kinect device. We also found that the precision of the Kinect (1-1.5 cm error tolerance) makes it suitable for gesture research.

We believe that the integration of motion capture data and the visualization of continous curves paves the way toward more objective coding scheme definitions and toward automated coding of motion. Moreover, it allows to compare modalities such as speech and motion both on the signal-level and on higher levels, all in a single tool.

Acknowledgements

Many thanks to Frederic Raber who conducted the Kinect precision experiments. This research has been carried out by the Embodied Agents Research Group (EMBOTS) at the German Research Center for Artificial Intelligence (DFKI), within the framework of the Excellence Cluster Multimodal Computing and Interaction (MMCI), Saarbrücken, sponsored by the German Research Foundation (DFG).

6. References

Jean Carletta. 1996. Assessing Agreement on Classification Task: The Kappa Statistics. *Computational Linguistics*, 22(2):249–254.

Alexis Heloir, Michael Neff, and Michael Kipp. 2010. Exploiting motion capture for virtual human animation: Data collection and annotation visualization. In *Proc. of the Workshop on "Multimodal Corpora: Advances in Capturing, Coding and Analyzing Multimodality"*.

Michael Kipp. 2001. Anvil – a Generic Annotation Tool for Multimodal Dialogue. In *Proceedings of Eurospeech*, pages 1367–1370.

Michael Kipp. 2012a. Anvil: The video annotation research tool. In Jacques Durand, Ulrike Gut, and Gjert Kristofferson, editors, *Handbook of Corpus Phonology*. Oxford University Press. to appear.

Michael Kipp. 2012b. Multimedia annotation, querying and analysis in anvil. In Mark Maybury, editor, *Multimedia Information Extraction: Advances in video, audio, and imagery extraction for search, data mining, surveillance, and authoring*, chapter 21. IEEE Computer Society Press.

Sotaro Kita, Ingeborg van Gijn, and Harry van der Hulst. 1998. Movement phases in signs and co-speech gestures, and their transcription by human coders. In Ipke Wachsmuth and Martin Fröhlich, editors, *Gesture and Sign Language in Human-Computer Interaction*, pages 23–35, Berlin. Springer.

Thomas Schmidt, Susan Duncan, Oliver Ehmer, Jeffrey Hoyt, Michael Kipp, Dan Loehr, Magnus Magnusson, Travis Rose, and Han Sloetjes. 2008. An exchange format for multimodal annotations. In *Proceedings of LREC*.

Peter Wittenburg, Hennie Brugman, Albert Russel, Alex Klassmann, and Han Sloetjes. 2006. ELAN: A professional framework for multimodality research. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*.