

Towards a Common Human Gesture Description Language

Mathias Kölsch & Craig Martell

Department of Computer Science, Naval Postgraduate School,
Monterey, CA 93943

Abstract

Body gestures are of tremendous importance for human-computer interaction, in particular for user interfaces for mixed reality. In this paper, we suggest a way to unify efforts from various fields to describe body language and gestures. First, taking multidisciplinary experiences into account, desired characteristics of such a “common gesture language” are laid out. We then propose a flexible language that allows greater decoupling of gesture description sources from their uses, for example, for interaction with virtual and augmented environments. The paper is meant as an informed starting point for discussion.

1. Introduction

Many, if not most, human-computer interaction tasks with mixed and augmented reality (MR) environments involve some form of manual input, where the human body actively influences the generated visuals and controls the application’s function. The use of a keyboard and mouse often does not lend itself well to facilitating this interaction – it may be better carried out with hand gestures, particularly in a mobile scenario [9]. In collaborative virtual environments, interaction additionally involves other people, their full body postures and facial expressions. MR technology and MR application developers currently must develop their own descriptions of humans interacting with and in their environments. Even though multiple methods for body tracking and posture estimation exist (e.g., motion capture and vision-based body tracking), they have different target audiences and are not directly interchangeable.

What is needed is a unified language capable of describing the human body to facilitate interaction in virtual environments (VE). This would help decouple the recognition technology from the VE developer. In this paper, we present existing descriptions for bodily gestures, find common traits, and identify the needs for a unification that is useful not only for mixed reality, but also for video annotation, psychological experiments, body language recognition, character animation, sign-language recognition and generation, and much more. Our goal is to bring all this under one umbrella in a systematic, extensible fashion and without creating a bloated super-language.

2. Related Work

For us a *gesture* is any bodily movement, voluntary or involuntary, or static pose of any body part, including the face. In the psychological and linguistic literature, a gesture is usually defined as voluntary expressions of meaning via bodily action.

This is to distinguish gestures from twitches, scratches and posture shifts that are not designed to convey meaning. In the computer vision literature, gesture usually refers to a continuous, dynamic motion, whereas a *posture* is a static configuration. As a common gesture language (CGL) needs to be as universal as possible, we draw upon both of these traditions.

In an early attempt to develop a coding scheme for gestural behavior, Efron [4] divided gestures into three distinct phases: preparation, stroke, and retraction. McNeill [11] and Kendon [8] augmented this analysis to develop full theories of how these phases relate to each other. For example, it is often the case that the retraction phase is omitted or blends smoothly into the next preparation phase. The FORM Gesture Annotation System [10] defines a *gesture excursion* to be all gestural movement between two rest positions of the arms, avoiding movement segmentation problems. Kahol et al. [6] automatically segment movement by combining coupled hidden Markov models and a hierarchical model of the body. A classification from Quek [12] focuses on non-speech-related gestures and adds unintentional movements and manipulative hand motions aside from communicative ones. Cadoz [2] distinguishes further between ergodic gestures – gestural *manipulation* – and epistemic gestures – tactile *exploration*.

Our notion of *gesture* includes movement of the face. Here, the Facial Action Coding System (FACS) [5] has produced significant results for both psychology and automatic facial-expression recognition [7]. It classifies facial expressions into qualitative “action units” (AU) such as raising the upper lip or parting the lips, without semantic meaning. Most AUs relate to a particular facial muscle and its movements. A *viseme* is the mouth shape made during pronunciation of a phoneme. Visemes are named facial expressions and can thus each be expressed as combinations of AUs.

Cal3D¹ is a library that implements many character animation tasks for rendering. As such it defines the notions of 1) a skeleton with bones and joints, 2) sets of vertices that are dependent in position on the skeleton, called a mesh, 3) material that is draped between the vertices, such as color and texture maps, and 4) animations that vary the skeleton’s joint angles over time. Similarly, the Humanoid Animation specification, or *h-anim*,² also describes the humanoid kinematics in form of a skeleton for animation purposes.

3. Requirements

In developing such a language, we should draw on the disciplines that have dealt with human body gesture description and use. First, there is a large body of psychology and language research aimed at describing gestures. Human factors, ergonomics, and biomechanics [13] extensively use body models such as Jack [3]. Computer graphics, especially animation and motion capture, have adopted some of these techniques. Computer vision, similar to psychology, is interested in infer-

¹<http://cal3d.sourceforge.net/>

²see <http://www.h-anim.org>

ring as much about the human body as possible from image sources. There, as in computer graphics, the models are slowly becoming more and more physically and anatomically correct as algorithms improve and computing power increases.

Given the varying needs of all these fields, a common gesture language (CGL) should permit:

- encoding of skeleton, muscles, and skin information;
- facial-expression encoding;
- descriptions of postures (static) and movements;
- syntax (e.g., 3D pos.) and semantics (ASL words);
- auxiliary data, such as the foreground region of the body, e.g., for occlusion rendering in MR;
- streaming/online and recording/offline analysis;
- a low data rate and/or compression;
- low-latency transmission, e.g., incremental updates;
- both absolute and relative time-stamping;
- synchronization of video and speech signals, e.g., visemes and phonemes;
- forward and inverse kinematics at information sinks;

3.1. Limitations

Not every body-related piece of information should be included in a CGL. For example, since no atomic elements are known a-priori, interpreters must ensure a grammatically correct order of events, not the language. Equally, the dynamics of articulated bodies and the interaction of limbs (causing occlusion, collisions, etc.) should be handled by translators and interpreters outside the language.

So far, the envisioned language describes only individuals. While multiple people can be described independently and unambiguously, no interactions between people is encoded. Thus, we may miss the significance of gestures directed at another person. Therefore, sign languages such as ASL can not be fully represented since they rely on many aspects of the physical environment, for example, to describe places and people that are not present. Similarly, eye-gaze is usually directed at the environment and frequently encoded in reference to it.

Every descriptor must be at least loosely associated with a body part. Unlabeled tracks of mo-cap markers, for example, are outside the scope of this CGL. Sensor movement needs to be described outside a CGL. Clock synchronization is also orthogonal and not part of a CGL. Lastly, we have not considered ways to describe clothing as well as type, style, shape, and movement of hair.

4. A Common Gesture Language

Satisfying all requirements described demands careful balancing between setting standards and allowing flexibility. Instead of defining atomic units of gestures, a possibly approach, we advocate that a common gesture language (CGL) should merely provide the framework that allows interested parties to define their own units, or *descriptors*, of gestures. Thus, we propose to standardize the way to specify a gesture descriptor with all its properties. Mutual information exchange in the

Table 1: Common gesture language’s descriptors: the fields (left column) and two examples: from a motion capture system, and the ASL sign for “eat,” in which the hand touches the open mouth. Matrix M_1 converts from the descriptor’s to the standard coordinate system.

descriptor name	MOCAP.left.wrist	ASL.eat
validity duration	.01s; $\sigma = 0s$.5s; $\sigma = .3s$
skeletal body parts	yes: left.wrist.subtree	indir: upper.body, yes:right.hand, yes:mouth (related)
motion type	posture	movement
spatial ref. frame	world	sternum
spatial datum	3D location	-
coord. transf.	M_1	-
spatial relation	-	touch
typ. accuracy, prec.	.2mm ³	-

same CGL is facilitated when every information source (capture) and information sink (interpreter, display, user interface) specifies through these descriptors what its capabilities are with respect to producible and usable gestures, respectively.

Crucial to this, a rather extensive, hierarchical skeleton should define a chain of joints with implicit interconnecting bones. The state of the skeleton is sufficiently described by the location of the joints, or, alternatively, by the state of the joints, which includes one or multiple angles. This skeleton need not be complete since extensions are systematically possible.

In the following subsections, we formalize the details of *descriptors* for gesture events. They represent the templates of events, that is, the communication partners’ dictionary. Next, we detail *events*, the actual bits of information that make up a stream of gestures.

4.1. Descriptors

A *descriptor*, published by an information source or sink to specify its language, contains a number of fields. Naturally, every descriptor has a unique **name**. To avoid ambiguity, a name should be part of a namespace identifying the source vendor, for example. The typical **validity duration** of the event is specified with mean and variance, where variable frame rate sources indicate this with a large variance. See Table 1 for examples.

The scope of **skeletal body parts** involved in the event is specified with terms from a kinematic hierarchy, the skeleton.³ Every body part needs to be assigned one of the following labels: not involved [default], yes=involved, and indirectly affected. The reason for distinguishing *not involved* and *indirect* is to provide interpreters with more knowledge about unspecified motion. For example, upper body sways during rigorous waving could be inferred through inverse kinematics in a smart renderer, and achieving motion blending for character animation becomes easier. The additional label *related* allows for gestures to be described as the interaction of two body parts. In the ASL example in Table 1, the mouth is directly involved

³see <http://www.movesinstitute.org/~kolsch/CGL/CGL.html>

(open), but also *related* to the hand’s movement.

The **motion type** of either *posture* (static) or *movement* distinguishes between an “infinitely short” sample point and an event of first- and second-order data, e.g., velocity and acceleration. This describes semantic duration, as opposed to the validity duration, which concerns the sample frequency. The **spatial reference frame** can be world-stabilized, relative to the event-producing sensor, or centered on a part of the skeleton.

OPTIONAL FIELDS:

Every descriptor can give at most one **spatial datum**, consisting of a 3D location and a 3D orientation (attitude), expressed in Quaternions. Additional data require their own event and/or descriptor.

Since there is no agreement on conventions for coordinate systems, not even within one discipline, no single system should be preferred over the others. The descriptor thus has to provide a **coordinate system transformation** to a common system, for example, right-handed, in meters:

x: East, or transverse right,

y: North, or antero-posterior forward, and

z: up, or longitudinal upwards.

This permits events to be described in the coordinate system native to the producer. If the space is discretized such as with FORM [10], this can be noted in a flag. Non-uniform discretization is not provided for and a suitable conversion must be employed outside the CGL context. This coordinate transformation can accommodate linear kinematic fitting, that is, adjustment of tracking or rendering models to the specific anthropometric data of a human. Non-linear and limb-specific kinematic fitting is currently not provided for.

Some sensors supply only reduced dimensionality data, for example, monocular camera systems natively provide 2D data. Thus, a second transformation has to describe the **sensor’s projection**, e.g., the projection matrix for camera systems.

In addition to FACS, we suggest **quantitative AUs**: the degree to which an AU is “deployed” with a) units normalized to [0; 1], where the binary values are the original FACS, b) the same coordinate system units as the skeleton, or c) units normalized to the anthropometric size of the head, which avoids an artificial “maximum deployment” of an AU and instead uses the anatomical limit.

Descriptors can specify **joint state** (extension/flexion, abduction/adduction) directly on the joints’ named axes of operation, as well as sliding motions as with the basal joint of the thumb. To allow encoding of more informal descriptions such as “pointing straight forward, slightly upwards,” the **spatial relationships** between multiple body parts can be described between the involved body parts as *touching*, at a certain *distance*, in a certain *direction*, and at an *angle* with respect to two involved body parts. For pointing, the hand’s location could be described by the angle between itself, the shoulder, and the hip joint (flexion) and the angle between itself and the two shoulder joints (abduction).

Different sensor types (frontal view, 3D data) will likely determine different classes of precision and accuracy that are pos-

sible, described in the **typical accuracy and precision** fields (in terms of the data fields (spatial location, orientation etc.).

A field for loosely **associated data** permits for structured description of information associated with some body parts, without rigidly formulating this associativity. Vertex meshes, which are computer animations’ way to add flesh, muscles, and skin to a skeleton, can be described this way. This field also allows for description of the data such as the appearance of the person in video and voxel data from stereo camera rigs. This is important for any system that only partially analyzes sensor data. Another example is video see-through mixed reality, which requires knowledge of hand and arm regions in the video in order to correctly omit rendering geometry that is occluded. The format of data in this field needs to be flexible, but multi-dimensional data that is aligned with the coordinate system strikes a good balance between flexibility and a standard.

4.2. Events

Here, we describe the fields of actual events, which have to be from a previously defined **descriptor type**. Next, a **person identifier** is necessary to distinguish events of multiple people. Conceivably, a globally unique ID could even be used to actually recognize people. A **timestamp**, taken at the start of the event, is important for non-real-time systems. It can be calendar time or relative to, e.g., the start of a video sequence. Often, translators will make correlations based on the temporal concurrence.

The **actual duration** of the event can be zero for snapshot-type events (to avoid the term “static”), such as motion capture data. It would be non-zero for *movement* motion types and indirectly determine the speed with which the movement occurred. It would also be non-zero for static postures that have persistence, such as sitting. The ending time of the event is timestamp plus duration.

The **probability** with which the event actually occurred can be 100%, but sources can also generate multiple events concurrently, each with probabilities of less than one. For example, an automatic translator might generate probabilities of less than one. The format in which to specify the **current accuracy** of the data depends on the type of data, such as 3D location or degrees.

Obviously, the real value of a language lies in someone understanding and correctly *interpreting* it. For most gesture languages, some interpreters do exist. So why do we need yet another language? The benefit of this CGL lies in the common way to define new “words” that can be used as part of the language from then on. This common way makes it much easier to write interpreters, pieces of software that understand a set of descriptors.

4.3. Translators

One particular kind of interpreters are **translators** that convert events from one set of descriptors into events from another set of descriptors. For example, “running” could be translated into a sequence of events from a different set of descriptors that

express more fine-grained notions of running, such as how the feet and hands move in a cyclic motion. In fact, that is what animation packages do: translate high-level motion descriptions into low-level skeleton joint motions.

A CGL facilitates the implementation of such translators since it uses the same syntax to define high-level and low-level descriptors. For animation, translators will often times require inverse kinematics to generate forward kinematics, that is, joint angles for end-effector positions. This is especially true for descriptors containing *relationships* of skeletal parts, such as the hand touching the nose.

Translators are not only important for top-down motion synthesis, but also for bottom-up motion and activity recognition, for example, for surveillance. An example of using a CGL for layered activity analysis is shown in Fig. 1. A grouping over time such as in annotation graphs [1] is a particular kind of translator.

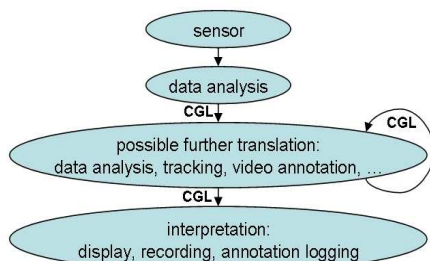


Figure 1: Gesture analysis, described in a CGL.

4.4. Format and Transmission

The exact syntax and format of the language is irrelevant to this effort. However, proven concepts such as XML are a likely candidate. If necessary, it can be converted into a binary format of smaller size, or simply compressed using one of the effective, XML-specific compression schemes. Note that the descriptors themselves are a high-level encoding method that can result in dramatic data size reduction.

Transmitting incremental updates to conserve bandwidth is also possible with stateful receivers. In that case, the transmitted fields of an event replace those of the previous event of the same descriptor type. For best-effort real-time streams, the timestamps can be approximated on the receiver side and need not be transmitted.

5. Discussion

We are debating the value of formalizing a descriptor hierarchy as part of the language (as opposed to implicit). For example, a parent descriptor type could exist for “hand shapes,” with child descriptors for fist, point, palm, etc., which would themselves be parents to a “left pinky metacarpophalangeal joint” descriptor.

Skin, muscles, and other non-skeletal information needs to be described as *associated data*. While sufficient for the purpose, a more rigorous descriptor field type should be sought to allow more straight-forward interpretation of the data.

6. Conclusions

We have described a procedure to define gesture events in a common, general framework. This promotes interoperability of such diverse fields as gesture recognition, motion capture, character animation, psychology, language research, and activity recognition. Event descriptors that use the same terminology even offer the potential of automatically generating translators that can convert between event streams at different semantic abstraction layers.

The goal of this document is not to impose a rigid standard. Instead, it is to be regarded as an initiator for discourse, for dialog between disparate communities, and for first ideas on how the wealth of information about the human body can be coded in a portable fashion. Instead of defining atomic units of gestures, we suggest a standard way to describe arbitrary units, which will then make up a particular vocabulary. We consider this approach more useful and also faster than trying to get every possibly interested party from industry and academia on the same table to define a standard. Still, their input has been and will be sought and is sure to contribute to shaping a common gesture language.

References

- [1] S. Bird and M. Liberman. A Formal Framework for Linguistic Annotation. Technical Report MS-CIS-99-01, U. of Penn., 1999.
- [2] C. Cadoz. Les réalités virtuelles, 1994.
- [3] N. B. Cary B. Phillips. Jack: A Toolkit for Manipulating Articulated Figures. In M. Green, editor, *Proc. ACM SIGGRAPH Symp. on User Interface Software*, pages 221–229, 1988.
- [4] D. Efron. *Gesture, Race, and Culture: A Tentative Study...* Mouton, The Hague, 1972.
- [5] P. Ekman and W. Friesen. The facial action coding system: A technique for the measurement of facial movement. *Consulting Psychologists Press*, 1978.
- [6] K. Kahol, P. Tripathi, and S. Panchanathan. Automated gesture segmentation from dance sequences. In *Proc. IEEE Intl. Conference on Automatic Face and Gesture Recognition*. IEEE, 2004.
- [7] A. Kapoor, Y. Qi, and R. W. Picard. Fully Automatic Upper Facial Action Recognition. In *IEEE Intl. WS on Analysis and Modeling of Faces and Gestures*, October 2003.
- [8] A. Kendon. An Agenda for Gesture Studies. *Semiotic Review of Books*, 7(3):8–12, 1996.
- [9] M. Kölsch, R. Bane, T. Höllerer, and M. Turk. Touching the Visualized Invisible: Wearable AR with a Multimodal Interface. *IEEE Computer Graphics and Applications*, 2006. in print.
- [10] C. Martell. FORM: An Extensible, Kinematically-based Gesture Annotation Scheme. In *Intl. Conf. on Language Resources and Evaluation*. European Language Resources Association, 2002.
- [11] D. McNeill. *Hand and Mind: What Gestures Reveal about Thoughts*. University of Chicago Press, 1992.
- [12] F. K. H. Quek. Eyes in the Interface. *Image and Vision Computing*, 13, August 1995.
- [13] G. Salvendy, editor. *Handbook of Human Factors and Ergonomics*. John Wiley & Sons, Inc, 2nd edition, 1997.