Coarticulation Analysis for Sign Language Synthesis

Lucie Naert, Caroline Larboulette, and Sylvie Gibet

IRISA, Université Bretagne Sud
Vannes, France
`{lucie.naert,caroline.larboulette,sylvie.gibet}@univ-ubs.fr`
`http://lsf.irisa.fr`

Abstract. A sign language utterance can be seen as a continuous stream of motion, involving the signs themselves and inter-sign movements or *transitions*. Like in speech, coarticulation constitutes an important part of the language. Indeed, the signs are contextualized: their form and, most of all, the transitions will greatly depend on the surrounding signs. For that reason, the manual segmentation of sign language utterances is a difficult and imprecise task. Besides, annotators often assume that both hands are synchronous, which is not always true in practice. In this paper, we first propose a technique to automatically refine the segmentation by adjusting the manual tags isolating signs from transitions. We then study motion transitions between consecutive signs and, in particular, the duration of those transitions. We propose several computation techniques for the transition duration based on the analysis we have conducted. Finally, we use our findings in our motion synthesis platform to create new utterances in French Sign Language.

Keywords: Coarticulation, Sign Language Segmentation, Transition Duration, Kinematic Features Analysis, Synthesis, French Sign Language (LSF)

1 Introduction

French Sign Language (LSF) is the natural language of deaf people in France and is therefore used as their first means of communication. As the linguistic mechanisms used in LSF are very different from those used in French, deaf people may face difficulties using written French. In LSF, the grammatical rules, as well as the nature and the spatial organization of the linguistic concepts are completely different from those used in French. However, most sources of information available on the Web and on other media are in French (written or oral), thus restraining Deaf access to information in their everyday life.

Virtual humans or *avatars* are a new and promising way to improve Deaf access to information. With avatar technologies, the anonymity of the signer is preserved. Furthermore, the content of sign language can be edited, manipulated, and produced more easily than with a video medium. Indeed, videos lack flexibility and operations such as copy/paste fail to deal with transitions in the context of editing new utterances.

Several approaches for synthesizing signed language utterances have been exploited over the years including keyframe techniques, procedural synthesis and data-driven synthesis. **Keyframe techniques** and **procedural synthesis** have been extensively developed in the international community. It allows a fine control at the language specification level which results in a precise behavior of the avatar. In return, the specification

step is fastidious and time-consuming. Moreover, the avatar may be poorly accepted by the deaf community due to the lack of realism of the resulting animations. To compensate for this realism issue, McDonald et al. [23] analyzed noise in motion capture data in order to extract linguistic and kinematic information. They injected this information into their keyframe-based synthesis tool to give their avatar a more life-like appearance. **Data-driven synthesis** often involves motion capture which is a very powerful tool to analyze movement features. Furthermore, as the synthesis is based on the movements of a real signer, the produced animations are more human-like. However, the capture of motion, the post-processing of data, the skeleton reconstruction and the annotation are costly both in time and resources. Besides, the corpus has to be large enough to account for the variability of sign languages.

Over the last ten years, we have worked on a data-driven approach for sign language synthesis. In our two last projects, *SignCom* [8] and *Sign3D* [9], we have developed a system in which we edit and compose new sentences that keep some linguistic coherence and are visually acceptable when played by an avatar. Some editing and animation issues have already been studied, implemented and discussed in [9]. Currently, our synthesis system relies on manually annotated data by expert annotators (deaf signers). However, manually segmenting sign language movements is a laborious and time-consuming process. Furthermore, this process depends on linguistic studies. In particular, the phonetic work proposed by Liddell and Johnson [15], as well as the annotation template proposed by Johnston and De Beuzeville [16] have largely inspired our annotation scheme in the *SignCom* project [8]. This template has been slightly refined in the *Sign3D* project to ease the annotation by avoiding the labeling of transition segments [9].

One of the main difficulties in the annotation process is that it is subject to variability due to the fact that all annotators do not agree with the starting and ending frames of the semantic segments [12]. Furthermore, as the signs in sequences appear in a continuous stream, one signer may start the subsequent movement before fully completing the previous one. This contextual dependency between signs, called *coarticulation*, makes the labeling more complex. Besides, the level of details of the annotation scheme greatly influences the way the avatar will be controlled.

This paper focuses on the analysis of inter-sign transitions to make the segmentation process as automatic and precise as possible, and to find motion invariants, in order to interactively improve the animation of the novel utterances signed by an avatar. Our final objective is to be able to incorporate coarticulation mechanisms into our concatenative synthesis system in order to generate a natural articulation between signs, while preserving the linguistic intelligibility. Two sub-challenges are considered in this paper: first, the segmentation of the transitions between signs is analyzed, leading to a refinement of the manual annotations; second, the duration of the transitions is deeply studied. The results are illustrated through interactive tools that are directly incorporated into our synthesis system.

2 Related Work

Segmentation of human motion is the process of breaking a continuous sequence of movement data into smaller and meaningful components, that range from actions to movement primitives. It is important here to emphasize that the segmentation may depend on its further use; in particular, this process is more constraining when the motion primitives relate to movement generation. The segmentation process consists in identifying the starting and ending frames for each segment corresponding to a movement primitive. The definition of the segments themselves is challenging due to the high-dimensional nature of human movement data and the variability of movement. For sign language movements, this is even more challenging since the segments depend on how the linguistic element boundaries are defined, according to phonetic, phonological and semantic rules, as well as coarticulation between signs. We review hereafter some segmentation work applied on general motion capture data and on sign language motion.

For general motion capture data segmentation, a frequently used approach is to identify segment boundaries by detecting sudden changes or threshold crossings in kinematic features, such as position and orientation and their derivatives (velocity, acceleration, curvature). For example, an indexing system has been developed by using joint positions relatively to other points or 2D planes [25] or by detecting the zero crossings in the angular velocity of the arm joints [6]. If these methods are easy to implement and can be efficiently applied on various data sets, they may give over or under segmentation boundaries and do not take into account semantic content.

Boundaries can be determined by other metric thresholds, using for example data analysis principles such as Principal Component Analysis (PCA), Probabilistic PCA, or Gaussian Mixture Models (GMM) [14], deriving Bayesian methods [5], or temporal application of Hilbert space embedding of distributions [10]. These methods, by projecting the motion capture data into low-dimensional representation spaces, give effective results, but fail to represent semantic data.

Other segmentation approaches use supervised learning techniques to take into account the semantic content of motion data. Among the proposed approaches, Müller et al. [24] used a genetic algorithm to identify characteristic keyframes while Brand et al. [2] used Hidden Markov Models.

Segmentation of sign language data has first been studied using video sequences. The complexity of the signs, characterized by many features (including hand movement, hand configuration and facial expression), requires the development of specific approaches. Lefebvre-Albaret et al. [21] developed a method to semi-automatically segment sequences of signs in French Sign Language (LSF) using region aspects: temporal boundaries are identified from a set of features including symmetry, repetition of movements, hand velocity and stability of the configuration. It introduces an interactive segmentation tool: first, a one-frame segment called *seed* is picked out by the user for each sign; this *seed* is then used by the region-detection algorithm to determine intervals containing all features of a sign. The resulting segmentation can be checked by a sign language annotator. Gonzalez and al. [11] have proposed to define an automatic two-level segmentation process: the first one, based on a robust tracking algorithm,

uses hand movement features; the second one uses hand configuration to correct the first level. A segmentation approach, based on a Hidden Markov Model and a state automata, has also been developed for Korean Sign Language segmentation [17]. Yang et al. [27] use Conditional Random Fields to distinguish transitions and signs in American Sign Language.

Coarticulation has given rise to a few works. Linguistically motivated models have been defined and validated through various approaches. Huenerfauth [13] has found that adding pauses and variations in sign durations improved the performance of virtual avatars by making the synthesized sentences more comprehensible by ASL signers. Pauses and temporal variations within signs were also introduced in LSF utterances by Segouat [26] to highlight the coarticulation effects in animations. A coarticulation model, incorporated into an inverse kinematic model has been developed for synthesis purpose [19]. Transitions have also been studied. A transition-inclusive system, separating the Strokes (S) conveying the meaning of the signs from the Preparation (P) and Retraction (R) phases characterizing inter-sign movements, has been initially proposed by [18] and used for LSF [1]. The transition shape has then been studied for recognition [20] or synthesis purposes [4].

Segmenting signs into significant components is still an open issue. Since the building of new sentences relies on the quality of the segmentation, we focus on this problem and we analyze inter-sign timing for synthesis purposes.

3 Our Synthesis System

Our synthesis system aims to build new utterances in LSF. To this end, it creates sentences concatenatively by:

(i) retrieving motion chunks corresponding to isolated signs, glosses (groups of signs with a specific meaning), or semantic components from our motion database, and

(ii) adding transitions between those chunks to create a continuous motion.

3.1   Data-Based Concatenative Synthesis

The system is based on a dual heterogeneous database containing both a database indexed by linguistic annotations and a second one indexed by the motion signal that has been recorded using motion capture. Furthermore, movements are annotated on different linguistic levels corresponding to grammatical indications, two-hand glosses, and finer phonological or phonetic levels (including right and left handshapes, hand placement and mouthing [9], [4]). These different levels are annotated on different channels.

To create a new LSF utterance, the user first edits the sentence by selecting a sequence of glosses organized in an order that is grammatically correct in LSF. The motion chunks corresponding to the glosses are then extracted from the motion database using the annotations. The concatenative synthesis then consists in concatenating those motion chunks by adding transitions between each glosse (see for example figure 1). This process is based on the assumption that the semantic annotation is accurate.

Note that in our system, we could query the database to retrieve motion primitives by channel, and we will show in section 4 that this would be more accurate, although the proposed synthesis presented at the end of the paper does not yet take advantage of this possibility.

Segmentation The aim of the segmentation is to explicitly identify at which frame a sign or a component of sign starts and at which frame it ends. One of the main difficulties of this process relies on the identification of the starting and ending of the meaningful part of the sign (called *meaning*).

Unlike speech in which transitions between words is silence, movement never stops and there is no reference starting point in space or rest hand configuration. However, there is a phase when the hand (or arm) prepares to move towards the starting position/configuration of the hand for the sign about to be executed. This phase is called *preparation* of the sign. For the same reason, before preparing to make the next sign, the hand/arm system first needs to retract to be ready to prepare for the next sign. This phase is named *retraction*. As we capture full sentences of LSF, each sign recorded in our database will be of the following form: *Preparation-Meaning-Retraction*. The annotation identifies the *meaning* part of each sign, the *preparation* and *retraction* being respectively the frames before and after the *meaning*.

Transition Synthesis We implemented two algorithms to synthesize transitions between two signs: the *Interpolation* and the *Blending* techniques.

The Motion Interpolation Method: it consists in doing a linear **interpolation** of both joint orientations and joint positions between the last posture of the first sign and the first posture of the second sign (see figures 1 and 13). In that case, we do not use the *retraction* motion of the first sign nor the *preparation* motion of the second sign. The duration of the interpolation (i.e. the number of frames n being added between both motion chunks) is based on the distance between M_{1_N} and M_{2_0} and the mean velocities of M_1 and M_2 (see section 5.2, SIMPLE_DURATION, for details on the computation). This method is simple to compute but does not take into account the transition data recorded in the database. As a result, it gives visually poor results when the duration exceeds a certain threshold.
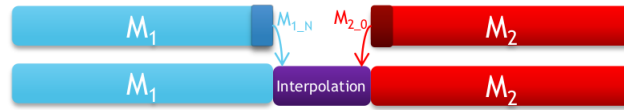

Fig. 1. Transition using the motion *Interpolation* method based on the last posture of the first sign and the first posture of the second sign.

The Motion Blending Method: it is computed as a linear **blending** of two motions: the movement following the first sign (*retraction* of M_1) and the movement preceding the second sign (*preparation* of M_2) in the *Sign3D* database (see figure 2). The transition length, i.e. the number of frames n used for the blending, is computed as previously

(see section 5.2, SIMPLE_DURATION, for details on the computation). This method gives better results in terms of realism (partial conservation of the context) and robustness with respect to a longer duration. However, the quality of the resulting movement greatly depends on the content of the database. For instance, if there is no captured motion before the second sign or after the first sign in the database, the transition will be less realistic (use of an idle skeleton with a default posture instead).
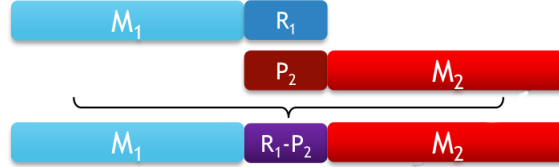

Fig. 2. Transition using the motion *Blending* method with the n first frames of the *retraction* phase R_1 of the first sign and the n last frames of the *preparation* phase P_2 of the second sign.

For both methods, the quality of the original annotation that will identify the *meaning* phase of a sign is of prime importance. However, when the annotation is manually done, usually by watching the corresponding video, it is not always easy to find the actual starting and ending frames of the *meaning* phase, let apart that both hands are not always synchronous as we will show in section 4.1.

The second important parameter of the concatenative synthesis is the choice of the length of the transition to be constructed. Considering the organization of the database, if the computed duration is too long, part of another sign could be extracted in addition to the *preparation* or *retraction* part. For example, in our database, a transition lasts on average about $0.30s$ (i.e. 30 frames) and never exceeds 50 frames while a sign may last about 20 frames. It is straightforward that if the computed transition time exceeds by 20 frames or more the transition time recorded in the database (> 50 frames), the transition will be computed using the previous sign and the *preparation* of the next sign when it should only use the *preparation* phase of the next sign. This results in a quite unrealistic motion that will appear as an hesitation from the avatar in the final synthesis. For the same reason, if the annotation is shifted by 10 frames from the actual point in time from where it should be, it will impact the final synthesis in the same way or by creating truncated motions.

This illustrates the need of a good computation of the transition duration as well as an accurate annotation of the database.

3.2   Overview of the Paper

In this paper, we first study the segmentation of the motion, by looking at the manual annotation provided by deaf experts in annotating LSF data as well as at an automatically refined annotation that we compute. Both annotations are then used to study the transitions in captured data to derive rules and metrics to use in our concatenative synthesis system to create new utterances in LSF (see figure 3).
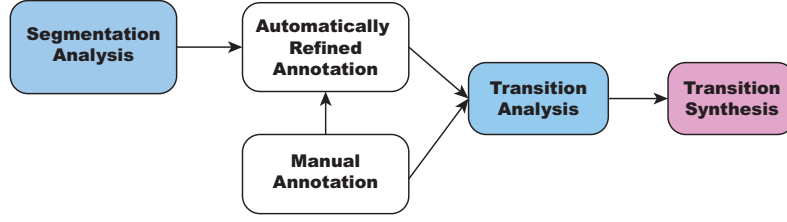
Fig. 3. Overview of the paper: we first study the segmentation of the captured data and improve the manual annotation; we then use both annotations to study the nature and length of transitions; we finally show some results using our findings in our synthesis system.

4   Segmentation

In sign languages, the signer alternates between signs and transitions. When processing recordings of sign language utterances, the first step consists in manually isolating signs from inter-sign movements using an annotation software such as ELAN [22, 3]. This step consumes both time and resources as at least one person has to review each video in order to produce a correct annotation. Another drawback of this manual annotation step is that it is a subjective task that greatly depends on the annotator's criteria and on the quality of the data to be processed. Indeed, it is often difficult to point out with certainty the beginning and end of a sign due to the inter-sign coarticulation and the continuous aspect of movement.

4.1   Kinematic Features of LSF Motion

In order to study a possible correlation between the kinematic properties of the hand motions and the sign/transition segmentation, sequences of LSF composed of two or three signs separated by transitions were examined. Those sequences, considered as the ground truth, are raw motions directly extracted from the *Sign3D* database which contains different utterances in French Sign Language. We have computed several kinematic features on those motions for various joints and observed that the speed (i.e. the norm of the velocity) of both wrists had interesting properties.

The figures 4 and 5 show the speed of both hands (left wrist and right wrist) for two different sign sequences. A green plain vertical line marks the beginning of a sign while a green dotted vertical line shows the end of the sign and the beginning of the transition, according to the manual segmentation. The dotted blue and red lines show the local minima of the two curves, respectively depicting the speed of the right hand and the left hand. The values were processed using a lowpass filter to prevent the algorithm from detecting all the incidental minima due to noise in the data.

According to the manual segmentation, the transitions seem to be delimited by two local minima in the norm of the velocity even though, due to the manual aspect of the task, the tags are not positioned exactly on the minima. Our hypothesis is thus that the correct segmentation of the signs should be on the local minima of the curves.

Furthermore, when doing manual segmentation, the whole body is considered. So, the starting and ending time of a sign is considered to be the same for all the skeleton
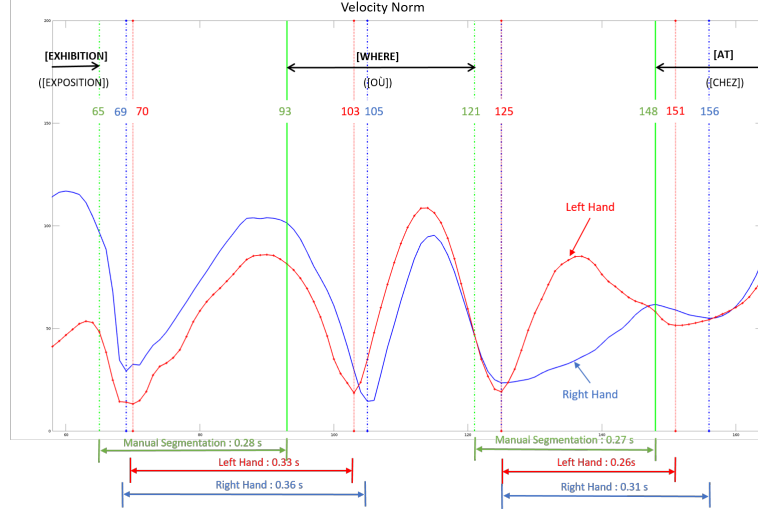
Fig. 4. Norm of the velocity of the left (red line with markers) and right (blue line) hands with respect to the frame number. The vertical lines show the edge of the transition defined by the annotator (green lines), the local minima of the right hand (blue lines) and the local minima of the left hand (red lines).

joints. However, that is not always true in practice. On figure 5, we can note that, assuming that minima delimit a transition, the transition of the left hand does not occur at the same time than the transition of the right hand. Indeed, there is an offset between the minima for each hand.

Therefore, we decided to refine the manual segmentation of motion capture data of French Sign Language by:

(i) detecting the speed minima of hand motions, and
(ii) assuming that the joints of the two hands are partially autonomous.

This refinement aims to improve the quality of the segmentation in order to perform a more accurate analysis and synthesis of LSF utterances.

4.2   Semi-automatic Segmentation

The detection of the speed minima of the hand motions is a fully automated process. However, it presents mixed results when used for LSF segmentation because of false positives when a sign is too noisy or complex (like the sign [AFTERNOON] which is a contraction of [AFTER] and [NOON] in LSF).

We have thus used a combination of the manual annotation and the local minima computation. When a minimum is close to a manual segmentation tag, the manual segmentation tag is replaced by the automatically computed minimum. If there is no minimum sufficiently close to a manual segmentation tag, it is kept with no modification (see figure 6: the minimum for the left hand ③ is too far away from the manual tag ②).
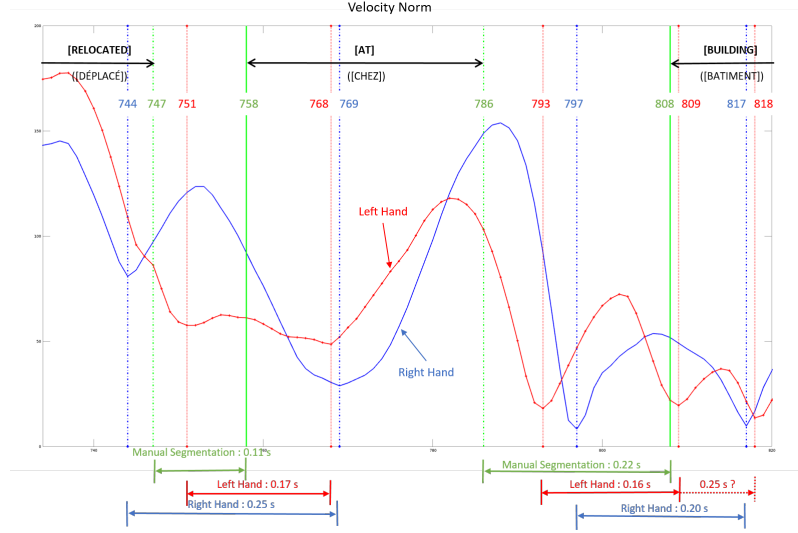
Fig. 5. Norm of the velocity of the left (red line with markers) and right (blue line) hands with respect to the frame number. The vertical lines show the edge of the transition defined by the annotator (green lines), the local minima of the right hand (blue lines), and the local minima of the left hand (red lines). An offset between the minima for each hand is visible.

We have used an empirical threshold of 15 frames for a capture frequency of $100Hz$: it corresponds to $0.15s$ of motion. A minimum distant of more than 15 frames from a manual segmentation is discarded. As the motion capture data was segmented by deaf people experts in annotating LSF data, we assumed that this relatively small threshold would result in a correct segmentation for a majority of cases. One benefit of this 15-frame threshold is that the minima of complex signs is removed from the segmentation (see figure 6, ④). This segmentation can be considered as an automated refinement of the manual segmentation that has the advantage of providing a different segmentation for each hand (see figure 6, ①).

5 Transition Duration

The length of transitions impacts the quality of synthesized animations of LSF utterances. A too short or too long transition will be perceived as strange and will often have repercussions on the general comprehension of the sentence [13]. The computation of a correct duration for transitions is therefore necessary.

5.1 Analysis

In order to define some empirical laws and invariants for the transition duration, we considered 89 transitions extracted from two sequences of motion capture data in the *Sign3D* database [9]. The first sequence (SIG_S1_X04) is a presentation of the opening
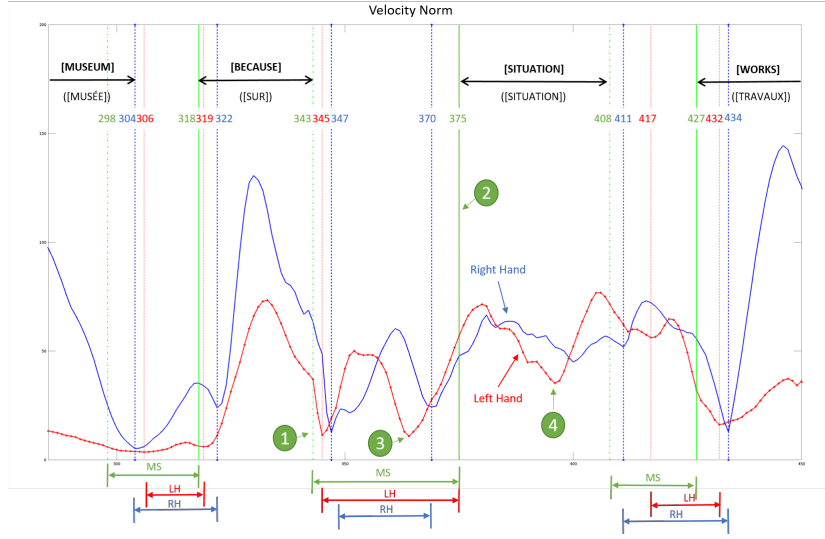
Fig. 6. Norm of the velocity of the left (red line with markers) and right (blue line) hands with respect to the frame number. The vertical lines show the edge of the transition defined by the annotator (Manual Segmentation (MS) with green lines) and refined by the segmentation procedure (blue lines for the right hand (RH) and red lines for the left hand (LH)).

and closing times of various town places (swimming pool, museum, etc.). The second sequence (SIG_S3_X02) explains the change of location of various exhibitions due to some incidents. Table 1 shows an example extracted from each sequence. The 89 transitions are the motions between two consecutive signs. They are extracted from the sequences using the annotations, either manual or the automatically refined annotation presented in section 4.2.

Table 1. Example of utterances contained in the database

Sequence	Extracted Utterances (*English translation*)
SIG_S1_X04	The swimming pool in front of the theater is open from 12:00 a.m to 11:00 p.m.
SIG_S3_X02	The museum exhibition was moved to the theater due to construction works.

Duration with Respect to the Distances Between Postures The **duration** of each transition was computed from the annotation times of the beginning tag and end tag of the transition.

The average distances between the last posture of the previous sign and the first posture of the next sign were computed. Two types of distances were used: the **Geodesic**

Distance between the joint orientations and the **Euclidean Distance** between the joint positions of the two skeletons. Each distance was averaged on the number of joints.

Considering two skeletons S1 and S2 composed of oriented joints, the Geodesic distance between the orientations (quaternions) of S1 and S2 is defined as the mean of the Geodesic distances between the orientations of each joint of S1 and the corresponding joint of S2:

$$\text{GeodesicDistance}(quaternion\ a, quaternion\ b) = ||log(a^{-1} * b)|| \qquad (1)$$

The Euclidean distance between S1 and S2 is defined as the mean of the Euclidean distances between the positions of each joint of S1 and the corresponding joint of S2:

$$\text{EuclideanDistance}(vector3d\ a, vector3d\ b) = ||a - b|| \qquad (2)$$

To take into account the two types of distances, we normalized them (by subtracting the mean value and dividing by the range of values) and computed the average distance.

Observations using Manual Annotations: figures 7 and 8 show the duration of the transitions with respect to the distances between the postures at the beginning and end of the transitions.
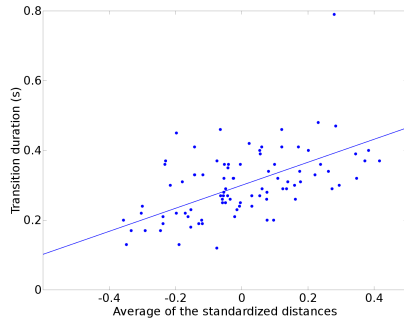

Fig. 7. Duration of the transitions in function of the average of the two normalized distances.
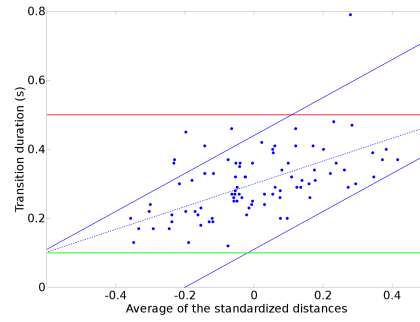
Fig. 8. More than 90% of the values is contained in the area bounded by the lines.

Considering our examples, we can note that:

1. The general tendency of the duration is to increase with the distance.
2. Apart from a single outlier, the duration never exceeds $0.5s$.
3. The duration never goes under $0.1s$.
4. More than 90% of the values is contained in a diamond shaped bounding box (see figure 8).
5. The mean duration of a transition is $0.303s$ with a standard deviation of $0.098s$.

Comparison with the Refined Segmentation (section 4): figure 9 shows the transition duration as a function of the mean Euclidean distance for the left joints (on the left) and for the right joints (on the right). The transition lengths of the left and right hand are considered independently and are, therefore, different for each hand. As each hand uses a different segmentation, only the arm joints (from the shoulder to the tip of the fingers) were studied and the mean Euclidean distance is, once again, computed between the last skeleton of the first sign and the first skeleton of the second sign. The blue diamond shaped markers represent the values for the manual segmentation whereas the red circle markers show the results considering the segmentation described in section 4.2. The lines between the pairs of markers depict the differences between the results of the two methods of segmentation.

Figure 9 confirms that there is an important difference, visible in the length of the lines connecting each two markers, between the segments defined by annotators and by the refinement using the velocity minima. However, the benefits of this new segmentation have to be ascertained using a quantitative and/or a perceptual evaluation.
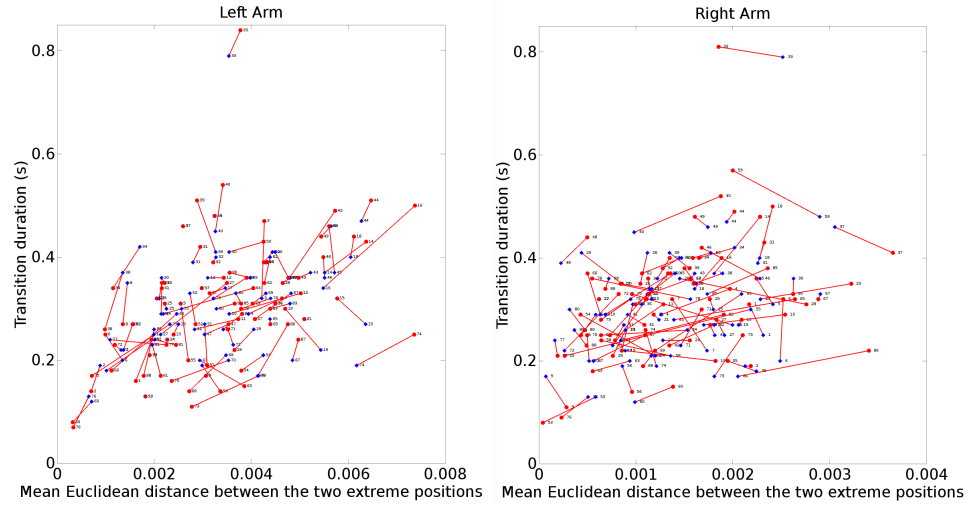

Fig. 9. On the left (right resp.): transition duration with respect to the mean of Euclidean distances of the left joints (right resp.) for the manual segmentation (blue diamond markers) and for the refined segmentation (red disk markers). The line between each pair of markers represents the difference in duration and distance between both segmentations.

Duration with Respect to the Type of the Surrounding Signs To determine if the length of the transitions is related to the nature of the surrounding signs, two different features of signs in LSF were examined in order to quantify their impact on the transition duration:

1. The number of hands used in the execution of the sign:
   - 0 = one hand ([MUSEUM] sign, for example),
   - 1 = two hands as in the [SWIMMING POOL] sign, and
   - 2 = one hand is doing a one-hand sign and the other is preserving the context: for example a [POINTING] gesture (one-hand sign) toward the other hand showing the remnant of the previous [EXHIBITION] sign (context). This is a case of contextualized signs.
2. The symmetry of the sign (only in the case of a two-hand sign). A sign is considered as symmetric if the two hands perform a symmetric motion:
   - 0 = the movement is not symmetric (e.g., [HOUR] sign), and
   - 1 = the movement is symmetric as in the [SWIMMING POOL] sign.

A transition was thus defined with four digits, the first two digits designating the features of the previous sign and the last two the features of the following sign. For example, a transition associated with the number 1011 is a transition between a two-hand asymmetric sign (10) and a two-hand symmetric sign (11). The transition 0010 is a transition between a one-hand and thus asymmetric sign (00) and a two-hand asymmetric sign (10). Table 2 lists all of the possible types of transition, their distribution in our data set and the mean duration and standard deviation according to the type of transition and the manual segmentation.

Table 2. List of all the transition types, mean duration and standard deviation

Transition Type	Number	Mean Duration (s)	Standard Deviation (s)
0000	6	**0.26833**	0.095167
0010	7	0.31000	0.090000
0011	5	**0.32000**	0.069642
0020	0	/	/
1000	9 (8)	0.36778 (0.3150 without outlier)	0.16292 (0.041057)
1010	9	**0.33444**	0.098629
1011	11	0.29273	0.054789
1020	0	/	/
1100	4	0.30500	0.077675
1110	16	**0.25813**	0.081912
1111	6	0.31667	0.14528
1120	6	0.29500	0.089610
2000	0	/	/
2010	1	0.30000	0
2011	5	0.30400	0.10761
2020	4	0.30250	0.078049

The shortest transitions are obtained for the transitions from two-hand symmetric signs to two-hand asymmetric signs (1110) and for the passage from one-hand signs to other one-hand signs (0000). Apart from the 1000 transitions whose result is impacted by the outlier (average without outlier: $0.3150s$), the longest transitions are between

two-hand asymmetric signs (1010). Figure 10 shows the mean duration of the transitions depending on the number of hands (type 2 considered as two-hand signs). The transition between two one-hand signs (1H → 1H) and two two-hand signs (2H → 2H) is shorter (and might be interpreted as easier) than adding or removing a hand between signs (1H → 2H and 2H → 1H). However, as the standard deviation is quite high compared to the mean values, the conclusion that can be made is that the number of hands and the symmetry of the signs surrounding a transition do not significantly impact the duration of the transition. An analysis of a higher number of transitions could lead to more conclusive results.
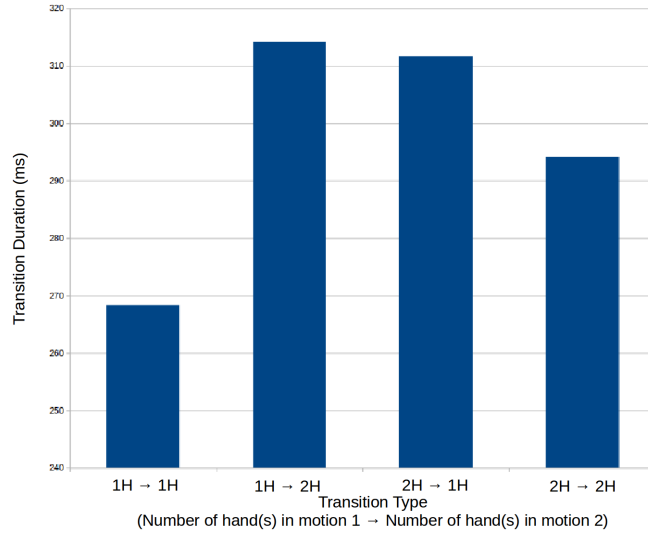

Fig. 10. Mean duration considering the number of hands involved in the first and second signs.

5.2   Synthesis

Computation of Transition Duration By using the results of our analysis, we aim to find a transition duration that best emulates the behavior of a real LSF signer in order to synthesize more natural and intelligible utterances.

SIMPLE_DURATION: previously to this analysis, the computation of the duration was based on the distance between the extreme positions *S1* (end of the first movement M_1) and *S2* (beginning of the second movement M_2) and the mean velocity of the two surrounding signs:

$$simpleDuration = (\alpha * \frac{2 * \text{EuclideanDistance}(S1, S2)}{MeanVel_{M_1} + MeanVel_{M_2}} +$$
$$(1 - \alpha) * \frac{2 * \text{GeodesicDistance}(S1, S2)}{MeanAngVel_{M_1} + MeanAngVel_{M_2}} \quad (3)$$

This method produces visually acceptable results for small distances and high velocities (short duration) but the computed transition duration is often longer than the ground truth equivalent. It reaches sometimes unacceptable values (sometimes as high as $1.5s$) that give unrealistic results with a slow down or an hesitation depending on the type of synthesized transition (*Interpolation* or *Blending* respectively).

Using the results of our analysis we propose three new computation methods of the transition duration.

SIMPLE_BOUNDED_DURATION: a first, simple measure is to put an empirical lower limit at $0.1s$ and a higher limit at $0.5s$ using the items #2 and #3 of the observations (section 5.1). The transitions with inconsistent duration are thus automatically changed to more correct values. We visually note an improvement in the rendering of the animation for the transitions involved.

LINEAR_DURATION: we also exploit the data of figure 7 to compute the coefficients of a trend line and use those coefficients to predict the value of a new transition with respect to the mean of the two normalized distances. In this way, we follow the Isochrony Principle (first mentioned in [7]).

SURFACE_DURATION: we compute the equation of a surface using normal equations to minimize the linear least square distance between the surface and the data (Euclidean and Geodesic distances of the 89 transitions). With this method, we find the optimal coefficients and use those to predict the duration of a new transition. Figure 11 shows the surface computed thanks to normal equations and the data that has been used to do the computation.

$$\theta = \left(X^T * X\right)^{-1} * X^T * y \tag{4}$$

- θ: the optimal parameters of the surface,
- X: a $89 * 3$ matrix containing the inputs (the first column contains only the value 1, the other two columns contain the Geodesic and Euclidean distances of each transition respectively), and
- y: a vector containing the durations of the transitions.

Using the equation (4) and our data, we found:

$$\theta = \begin{pmatrix} 0.148777 \\ 0.663958 \\ 0.275229 \end{pmatrix}$$

The mean error between the real and computed duration for our 89 transitions can be calculated as:

$$MeanError = \frac{1}{89} \sum_{i=1}^{89} \sqrt{(y_i - (X_i * \theta))^2} = 0.06075s \tag{5}$$

As the analysis of section 5.1 concluded that the number of hands and symmetry of the surrounding signs did not have a great impact on the transition duration, these two parameters were not taken into account in the duration computation.
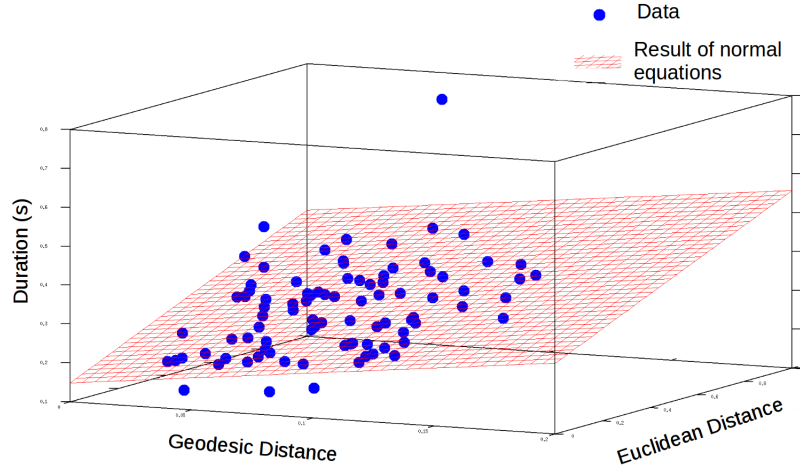
Fig. 11. Surface computed using normal equations (4) with the manual annotation data.

Results In this section, we compare and visualize the results of the various computations methods of the transition duration.

We first compare **synthesized transitions with their ground truth equivalent** on a limited number of examples. The ground truth is the original motion capture signal segmented according to the manual annotation. The transition synthesis methods (*Interpolation* or *Blending*) and the duration computation methods (XXX_DURATION) are tested by comparing the resulting animation with the ground truth equivalent. Figure 12 illustrates the transition between the signs [BUILDING] and [THEATER]: an offset between the two skeletons is visible when playing the animation. This offset changes with the parameters of the transition. While the duration of the ground truth is of $0.27s$, the LINEAR_DURATION method gives the closest result with a duration of $0.30s$ whereas the SIMPLE_DURATION method is the farthest with $0.37s$.

Then, we compared the **synthesized transitions with each other** by choosing to pair any, not necessarily consecutive, signs in the database. It is thus impossible to compare the performances of the generated transition with the ground truth which does not exist but, instead, we can compare the synthesized transitions with one another.

On figure 13, the user has chosen to analyze the transition between the sign [TO PAY] and the sign [MUSEUM]. The duration of the corresponding transition has been computed for each of the methods. We can see that the SIMPLE_DURATION method gives an abnormally high value of $1.47s$. Indeed, the Euclidean distance between the two extreme skeletons of the transition is equal to 1.08172 and the Geodesic distance is 0.103753. The high value of the Euclidean distance can be explained by the fact that [TO PAY] is a two-hand sign and [MUSEUM] is a one-hand sign (see figure 13).
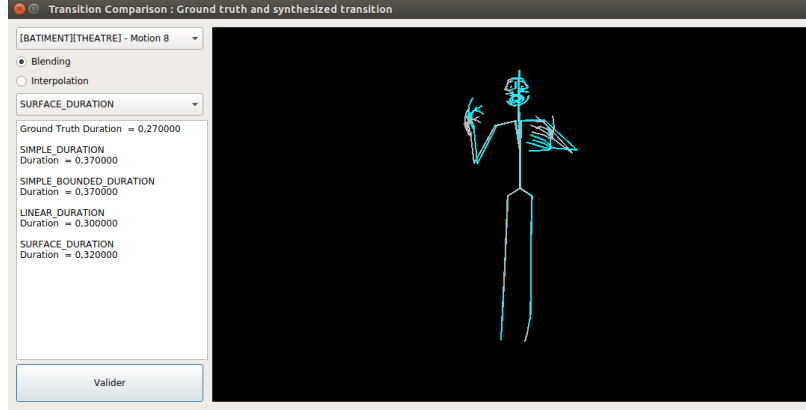
Fig. 12. Comparison of the ground truth with the synthesized transitions. The two skeletons representing the ground truth (in white) and the synthesized motion (in cyan) are superimposed.

The resulting transition is not convincing using either the *Interpolation* (slowdown) or *Blending* (artifacts due to unwanted sign chunks added to the animation) generation techniques.

The SURFACE_DURATION gives a much more acceptable result with a transition of $0.51539s$:

$$duration = \theta_0 + \theta_1 * GeoDist + \theta_2 * EucDist$$
$$= 0.148777 + 0.663958 * 0.103753 + 0.275229 * 1.08172$$
$$= 0.51539s$$

The other methods do not allow the duration to exceed the $0.5s$ boundary and their results are also more convincing than the SIMPLE_DURATION method.

Improvement of Motion Retrieval The previous analysis shows that the distance between two consecutive motions has an impact on the transition duration. We also know that a long duration will often be less realistic than a shorter duration. We thus implemented a new motion retrieval method based on those observations. This method retrieves a motion from a range of possible motions in order to minimize the average distance of this motion with the previous and/or following motion. By minimizing the distance, the duration of the transition is naturally shorter than if we had taken a random motion among all of the possible motions.

6   Conclusion

In this paper, we studied the coarticulation aspect of sign languages by focusing on the transition segmentation and duration.

We first proposed a new way of segmenting LSF utterances by analyzing some kinematic properties of sign language motions and, more specifically, by detecting local
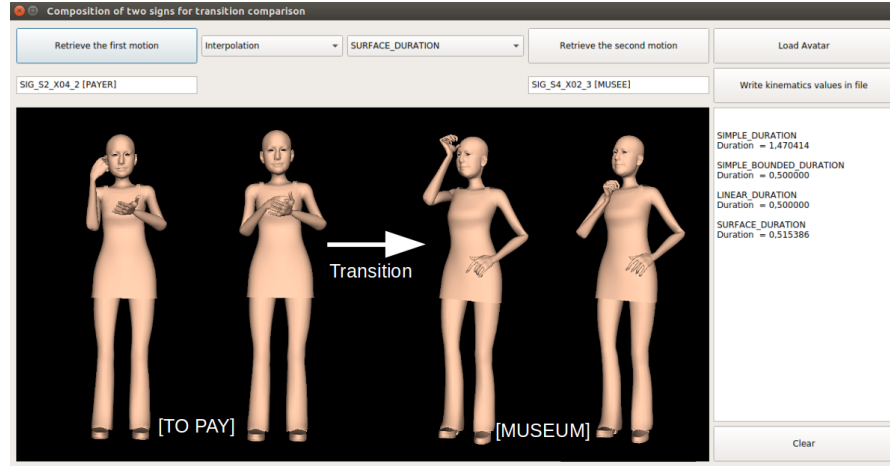
Fig. 13. Composition of two signs for transition comparison. The last posture of the first movement is quite far from the first posture of the second movement.

minima in the norm of the velocity for each hand. The manual annotations done by deaf experts were then refined by selecting the nearest corresponding minima for each manual segment.

In a second phase, the transition duration was analyzed with respect to the distance covered during a transition and to the type of the surrounding signs. Based on our observations, we defined new methods to compute transition durations based on our observations. An extremely simple and effective method resulting in a visual improvement of the animation is to introduce a lower and upper limit in the duration. Other approaches using basic statistics on the data were implemented. For our dataset, the nature of the surrounding signs did not impact significantly the duration of the transition. An analysis of a higher number of transitions could lead to more conclusive results. This approach raises questions concerning the synchronization of the sign language channels. Indeed, our segmentation is based on the observation that each hand has a partially autonomous behavior. The offset between both hands will be the focus of future studies.

Using these observations, we aim to improve our synthesis engine by handling the two hands separately. This process will be evaluated thanks to perceptual studies with native LSF signers testing, on the one hand, the intelligibility and comprehensibility of the produced sentences and, on the other hand, the acceptability and realism (3D rendering, smoothness, etc.) of the animation. Other issues may be addressed such as the advantages of segmenting each channel (hand configuration, placement, orientation, non-manual features, etc.) separately both for retrieving signs in our database and for synthesizing sentences. In the future, we would like to use our semi-automatic segmentation on a larger corpus composed of carefully chosen utterances to perform further analyses.

References

1. C. Awad, N. Courty, K. Duarte, T. Le Naour, and S. Gibet. A combined semantic and motion capture database for real-time sign language synthesis. In *IVA*, pages 432–438, 2009.

2. M. Brand and V. Kettnaker. Discovery and segmentation of activities in video. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):844–851, August 2000.

3. O. Crasborn and H. Sloetjes. Enhanced ELAN functionality for sign language corpora. In *Proceedings of LREC 2008, Sixth International Conference on Language Resources and Evaluation*, 2008.

4. K. Duarte. *Motion Capture and avatars as Portals for Analyzing the Linguistic Structure of Sign Languages*. PhD thesis, Université Bretagne Sud, 2012.

5. D. Endres, A. Christensen, L. Omlor, and M.A. Giese. Segmentation of action streams human observers vs. bayesian binning. In *Proceedings of KI 2011: Advances in Artificial Intelligence, 34th Annual German Conference on AI*, pages 75–86, 2011.

6. A. Fod, M.J. Matarić, and O.C. Jenkins. Automated derivation of primitives for movement classification. *Autonomous Robots*, 12(1):39–54, 2002.

7. F.N Freeman. Experimental analysis of the writing movement. *The Psychological Monographs*, 17:1–57, 1914.

8. S. Gibet, N. Courty, K. Duarte, and T. Le Naour. The *SignCom* system for data-driven animation of interactive virtual signers: Methodology and evaluation. In *Transactions on Interactive Intelligent Systems (Tiis)*, volume 1, pages 6:1–6:23, 2011.

9. S. Gibet, F. Lefebvre-Albaret, L. Hamon, R. Brun, and A. Turki. Interactive editing in French sign language dedicated to virtual signers: requirements and challenges. *Universal Access in the Information Society*, 15(4):525–539, 2016.

10. D. Gong, G.G. Medioni, S. Zhu, and X. Zhao. Kernelized temporal cut for online temporal segmentation and recognition. In *Proceedings of the 12th European Conference on Computer Vision ECCV - Part III*, pages 229–243, 2012.

11. M. Gonzalez. Un système de segmentation automatique de gestes appliqué à la langue des signes (French) [An automated gesture segmentation system applied to sign language]. In *JEP-TALN-RECITAL*, page 9398, 2012.

12. T. Hanke, S. Matthes, A. Regen, and S. Worseck. Where does a sign start and end? Segmentation of continuous signing. *Language Resources and Evaluation Conference*, 2012.

13. M. Huenerfauth. A linguistically motivated model for speed and pausing in animations of american sign language. *ACM Trans. Access. Comput.*, 2009.

14. J.Barbic, A. Safonova, J.-Y. Pan, C. Faloutsos, J.K. Hodgins, and N.S. Pollard. Segmenting motion capture data into distinct behaviors. *Graphics Interface*, May 2004.

15. R.E. Johnson and S.K. Liddell. A segmental framework for representing signs phonetically. *Sign Language Studies*, 11(3):408–463, 2011.

16. T. Johnston and L. De Beuzeville. Researching the linguistic use of space in auslan: Guidelines for annotators using the auslan corpus. In *Technical report, Department of Linguistics, Macquarie University*, 2009.

17. J.-B. Kim, K.-H. Park, W.-C. Bang, and Z.Z. Bien. Continuous korean sign language recognition using gesture segmentation and hidden markov model. *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, 2001.

18. S. Kita, I. van Gijn, and H. van der Hulst. Movement phase in signs and co-speech gestures, and their transcriptions by human coders. In *Proceedings of the Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction*, volume 1371 of *Lecture Notes in Computer Science*, pages 23–35. Springer-Verlag, London, 1997.

19. T. Lebourque and S. Gibet. High level specification and control of communication gestures: the GeSsyCa system. In *Proc. of Computer Animation*, Genova, Switzerland, May 1999.

20. F. Lefebvre-Albaret. *Traitement automatique de vidéos en LSF. Modélisation et exploitation des contraintes phonologiques du mouvement (French) [Automatic processing of LSF videos. Modelling and exploitation of the phonological constraints of motion]*. PhD thesis, Université Paul Sabatier - Toulouse III, 2010.

21. F. Lefebvre-Albaret, P. Dalle, and F. Gianni. Toward a computer-aided sign segmentation. *Language Resources and Evaluation Conference (LREC). European Language Resources Association*, 2008.

22. Max Planck Institute for Psycholinguistics, The Language Archive, Nijmegen, The Netherlands. http://tla.mpi.nl/tools/tla-tools/elan/.

23. J. Mcdonald, R. Wolfe, R.B. Wilbur, R. Moncrief, E. Malaia, S. Fujimoto, S. Baowidan, and J. Stec. A new tool to facilitate prosodic analysis of motion capture data and a data- driven technique for the improvement of avatar motion. In *Proceedings of the 7th Workshop on the Representation and Processing of Sign Languages: Corpus Mining Language Resources and Evaluation Conference (LREC)*, volume 7, 2016.

24. M. Müller, A. Baak, and H.-P. Seidel. Efficient and robust annotation of motion capture data. *Symposium on Computer Animation*, 2009.

25. M. Müller, T. Röder, and M. Clausen. Efficient content-based retrieval of motion capture data. *ACM Trans. Graph.*, 24(3):677–685, 2005.

26. J. Ségouat. *Modélisation de la coarticulation en Langue des Signes Française pour la diffusion automatique d'informations en gare ferroviaire l'aide d'un signeur virtuel. (French) [Modelling coarticulation in LSF for automatic broadcast of information in train stations using an avatar]*. PhD thesis, Université Paris Sud - Paris XI, 2010.

27. R. Yang and S. Sarkar. Detecting coarticulation in sign language using conditional random fields. In *Proceedings of the 18th International Conference on Pattern Recognition*, volume 2, pages 108–112, Washington, DC, USA, 2006. IEEE Computer Society.