Eurographics/ ACM SIGGRAPH Symposium on Computer Animation (2006)
M.-P. Cani, J. O'Brien (Editors)

Automatic Splicing for Hand and Body Animations

A. Majkowska[1], V. B. Zordan[2] and P. Faloutsos[1]

[1]University of California, Los Angeles
[2]University of California, Riverside

Abstract

We propose a solution to a new problem in animation research: how to use human motion capture data to create character motion with detailed hand gesticulation without the need for the simultaneous capture of hands and the full-body. Occlusion and a difference in scale make it difficult to capture both the detail of the hand movement and unrestricted full-body motion at the same time. With our method, the two can be captured separately and spliced together seamlessly with little or no user input required. The algorithm relies on a novel distance metric derived from research on gestures and uses a two-pass dynamic time warping algorithm to find correspondence between the hand and full-body motions. In addition, we provide a method for supplying user input, useful to animators who want more control over the integrated animation. We show the power of our technique with a variety of common and highly specialized gesticulation examples.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computing Methodologies]: Computer GraphicsThree-Dimensional Graphics and Realism: Animation

1. Introduction

Gesticulation is an important aspect of human motion. Beyond grasping and manipulation tasks, gestures are commonly used in conversation to convey information; precise hand movements are an important component of many dance styles and performances. In computer animation, hand motion is typically either keyframed by an animator or added painstakingly by hand, starting from motion capture data recorded during dedicated hand capture sessions.

While the development of methods for re-use and modification of motion capture data is an active area of research, little attention has been paid to automatic methods for integrating hand motion into full-body animations. Although it is possible to record hand movement together with full-body motion in a simultaneous capture, there are compelling reasons for recording hand movement separately. These include greater flexibility in the use and re-use of hand motions across actors and animated scenes, and increased accuracy and control over the hand capture. The latter is especially true for optical technologies, where resolution and occlusion are key factors in the quality of the captured motion.

We present a technique that seamlessly splices together

Figure 1: *Mudras in Indian dancing.*

hand and body motions recorded in separate sessions, while maintaining synchronization between the two motion capture sources. Finding the correct matching between hand and body motions recorded separately can be a challenging problem due to differences in motion execution. In particular, timing and amplitude variations can cause two recordings of the same movement to have very different numerical representations. Differences in timing arise when parts of two motions are performed at different speeds. Amplitude variations are caused by different execution of the motion, for example, a performer extending his arm further in one performance over another. This problem is prevalent when high resolution hand motion is recorded with optical systems because the motion is often constrained to a small, restricted area to avoid marker occlusion.

Despite the timing and amplitude variations, humans can easily recognize and interpret different executions of the same hand movement. Research on human gestures shows that the hand movement can be segmented into distinct phases using objective measures [KvGvdH98]. Specifically, *phase boundaries in hand movement are marked by an abrupt change of direction with a discontinuity in velocity profile before and after the direction change.* Exploiting this criteria, we propose a novel distance metric that assesses the signs of the first and second derivatives of motion trajectories over time in order to detect phase boundaries.

To find correct correspondence between hand and full-body motions, we use our metric for hand motion segmentation jointly with dynamic time warping (DTW). Our use of DTW for such a task is novel because, although DTW has been introduced to the graphics community previously for temporal alignment, the extreme amplitude differences seen between performances in our case have not been addressed in previous work.

We employ the DTW algorithm at two levels of refinement. First, we use it at the coarse level to identify phase similarity. By aligning phases of motion our method overcomes gross amplitude and timing differences. Next, our algorithm performs a second DTW pass within each matched phase in order to fine tune timing correspondence. We extended the basic DTW algorithm to allow for user input in order to guide the splicing process.

The contributions of this paper are:

- **A solution to a new problem of adding gesticulation into full-body animation.** Occlusion and a difference in scale make it difficult to capture both the detail of the hand movement and unrestricted full-body motion at the same time. We offer a solution in which the two can be captured separately and spliced seamlessly together with little or no user input required. Our algorithm can be also used to enhance existing animations by incorporating hand motion with the same ease.
- **A simple and effective technique for hand and body motion alignment.** Our method uses dynamic time warping together with a novel distance metric to align motions with significant amplitude differences. Key features about human gestures, that allow our algorithm to find proper alignment, create the foundation for our distance metric and are supported by literature that comes directly from researchers who study gesture.
- **A method for incorporating user-specified constraints directly into the DTW algorithm.** Our method allows a user to easily choose from a continuous spectrum of control options: from fully automatic matching with no user input, to partial control, where the user suggests matching regions, to complete control, with the user listing specific pairs of frames to be matched.

2. Related Work

Researchers have introduced numerous techniques for editing and combining human motion data with emphasis placed on re-use and generalization for animation applications. Many of the described methods focus on transforming individual motion clips in order to adjust the behavior recorded within the given data segment, for example [WP95, Gle97, PW99, LS99, SP05]. Other approaches address the problem of joining multiple motion segments, offering solutions to problems associated with automatically creating *transitions* between motions in large databases [AF02, KGP02, LCR*02] and for generating visually pleasing transitions between two clips [RGBC96, KG03]. A third popular topic includes building a continuous space of poses or motions from a finite set of examples, often using a simple set of input parameters in order to control aspects of the poses or resulting animations, in [WH97, RCB98, GSKJ03, GMHP04, HGP04, KG04, SHP04, CH05].

A more narrow class of techniques targets the combining of motion samples which overlap in time. While the previously described approaches combine cuts made along the time axis, fewer examples in the literature describe methods for layering motions that occur simultaneously, and most of this work has been based on extracting style elements from one motion and layering them over another [UAT95, RCB98, PB00, BH00, SCF04, HPP05].

We aim our efforts at automatically combining different motion clips onto a single skeleton by allowing each clip to control its own portion of the skeletal hierarchy. The work of Ikemoto et al. [IF04] shares the characteristic of integrating two motions, which control different parts of the character skeleton, into a single hierarchy. Our work is unique in that we are interested in combining recordings from different sources by finding correlations within the motion clips and synchronizing the timing of the clips to align them in a coherent manner. In their work, Ikemoto et al. transplant limbs from one clip to another by selecting random pairs from a library and forming an arbitrary transplantation and then accessing the quality based on a set of rules. In con-

trast, we focus on combining known, user-specified clips, and place our efforts in the careful alignment over time of the motions contained in the given sources. Our work is also similar to that of Dontcheva et al. [DYP03] in our effort to align motion sequences recorded in separate motion-capture takes and to apply them to a single skeleton. However, our approach, driven by the specific problem domain, accounts for amplitude differences that change over time in a non-uniform and non-discrete manner. This is crucial in the area of hand gesticulation, where different recordings of the same gesture sequence tend to have large, varying amplitude differences.

Although other researchers have used time warping for motion alignment, for example [KG03, KG04, FF05], we applied it to a novel domain of human gesticulation. This posed new challenges to our system. While existing work concentrated on more dynamic motions, such as walking or jogging, human gesticulation tends to have more kinematic qualities with significantly more variation than motions controlled to a large extent by physical constraints. Because of these differences, previously proposed comparison metrics do not work well in the domain of gesticulation. Additionally, although the idea of a user controlling the alignment by specifying pairs of matching frames was used before (for example by Rose et al. [RGBC96]), we are the first ones to incorporate the constraints directly into the DTW band in order to allow for a varying degree of user control.

3. Matching Algorithm

Our technique employs dynamic time warping (DTW) to generate frame correspondences between motions with timing differences. Our algorithm works in three stages: first, using time warping, we match movement phases based on the velocity and acceleration profile. Next, we modify the resulting match by adjusting the alignment of the frames within the phases, again with DTW. Finally, we create the combined hand and body animation, smoothing the resulting motion where necessary.

To match our source motions, we apply DTW with distance functions computed from four marker positions (two on the wrist, one on the hand, and one on the forearm). We found that this set was typically easy enough to include in both full-body and hand motion data during our captures, however the distance function can easily be adapted for use with different markers or with joint angle data.

Dynamic time warping: For detailed description of the time warping technique see [KR05]. Briefly, given the sequences $\mathbf{p} = \{p_1, p_2, \ldots, p_n\}$ and $\mathbf{q} = \{q_1, q_2, \ldots, q_m\}$, the DTW algorithm computes a cumulative distance matrix $\Gamma = \{\gamma_{i,j}\}_{1,1}^{n,m}$ using a recurrence equation:

$$\gamma_{i,j} = R(i, j, \gamma_{i-1,j-1}, \gamma_{i-1,j}, \gamma_{i,j-1}) \tag{1}$$

where

$$R(i, j, v_1, v_2, v_3) = \mathbf{D}(p_i, q_j) + \min \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \tag{2}$$

and $\mathbf{D}(p_i, q_j)$ is the distance between points p_i and q_j. $\gamma_{i,j}$ represents the the minimal cost of a monotonic path between points $(0,0)$ and (i, j). This path defines the correspondences between the frames of two motions, where each frame of one motion is matched to one or more frames of the other.

In order to limit the number of frames that can be matched to one frame and to reduce the computation cost, the search for an optimal path is typically limited to a restricted area of the matrix, called the *band* (see Figure 2(a)). To incorporate the band constraint, we compute a new matrix Γ' as follows:

$$\gamma'_{i,j} = \begin{cases} R(i, j, \gamma'_{i-1,j-1}, \gamma'_{i-1,j}, \gamma'_{i,j-1}) & \text{if } j \geq \frac{m}{n} \cdot i - \Delta, \\ & \quad j \leq \frac{m}{n} \cdot i + \Delta \\ +\infty & \text{otherwise} \end{cases} \tag{3}$$

where $\Delta = W \cdot \frac{\sqrt{n^2+m^2}}{n}$ and W denotes the width of the band.

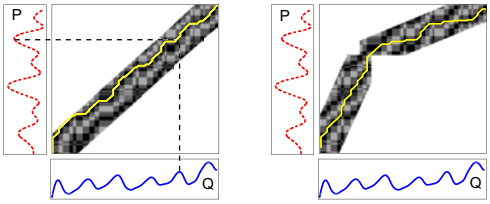
We extend the DTW algorithm to add functionality for user input; a user can specify which pairs of frames in the two motions should be matched and the algorithm will adapt the search band to the user's specifications (see Figure 2(b)). A user can choose if frames need to be matched exactly or with some tolerance; in the latter case the search band will also include neighboring frames. We need to further modify the equation (3) to accommodate user specified constraints, which are represented as a set of matching points $(x,y)_{k=1}^{l}$, where x_i and y_i are frame indexes in $\mathbf{p}$ and $\mathbf{q}$ respectively, $x_0 = 0 < x_1 < x_2 < \ldots < x_l < x_{l+1} = n$ and $y_0 = 0 < y_1 < y_2 < \ldots < y_l < y_{l+1} = m$. First we need to modify the band shape, so it includes the matching points:

$$\gamma''_{i,j} = \begin{cases} R(i, j, \gamma''_{i-1,j-1}, \gamma''_{i-1,j}, \gamma''_{i,j-1}) & \text{if } j_k \geq \frac{|y_k|}{|x_k|} \cdot i_k - \Delta_k, \\ & \quad j_k \leq \frac{|y_k|}{|x_k|} \cdot i_k + \Delta_k \\ +\infty & \text{otherwise} \end{cases} \tag{4}$$

where k is an index s.t. $x_{k-1} < i \leq x_k$, while $i_k = i - x_{k-1}$, $j_k = j - y_{k-1}$, $|x_k| = x_k - x_{k-1}$, $|y_k| = y_k - y_{k-1}$ and $\Delta_k = W \cdot \frac{\sqrt{|x_k|^2+|y_k|^2}}{|x_k|}$.

Next, we limit the band area so that the minimal path is constrained to pass in the neighborhood of the points (x_k, y_k), within the user-specified tolerance T. We compute the final matrix Γ^* as follows:

$$\gamma^*_{i,j} = \begin{cases} R(i, j, \gamma^*_{i-1,j-1}, \gamma^*_{i-1,j}, \gamma^*_{i,j-1}) & \text{if } lb \leq j_k \leq ub \\ +\infty & \text{otherwise} \end{cases} \tag{5}$$



(a) (b)

Figure 2: *Dynamic time warping for motions P and Q. (a) We compute the distance matrix for pairs of frames from P and Q. We limit our search to a portion of distance matrix called a band (diagonal region). The darker areas on the band denote lower cost. The bright path denotes the optimal alignment between frames that minimizes the total cost. (b) Our technique allows for user input by modifying the band. A user can specify pairs of matching frames either exactly (band limited to one point) or with some tolerance (band limited to a narrow area).*

where *lb* and *ub* are defined as:

$$lb = \max\{1, y_{k-1} - T, \frac{|y_k|}{|x_k|} \cdot i_k - \Delta_k\}$$

$$ub = \min\{m, y_k + T, \frac{|y_k|}{|x_k|} \cdot i_k + \Delta_k\}.$$

The above formula can be easily transformed into an efficient scheme for computation of matrix Γ^* and finding optimal alignment. Algorithm 1 shows a possible implementation.

Algorithm 1 Compute matrix Γ^*

```
initialize all elements of Γ* to +∞
for all k such that 0 < k ≤ l + 1 do
  for all i such that x_{k-1} < i ≤ x_k do
    |x_k| = x_k − x_{k-1}, |y_k| = y_k − y_{k-1}
    Δ = W · sqrt(|x_k|^2 + |y_k|^2) / |x_k|
    i_k = i − x_{k-1}
    lb = max{1, y_{k-1} − T, (|y_k|/|x_k|) · i_k − Δ_k}
    ub = min{m, y_k + T, (|y_k|/|x_k|) · i_k + Δ_k}
    for all j such that lb ≤ j − y_{k-1} ≤ ub do
      γ*_{i,j} = D(p_i, q_j) + min [ γ*_{i-1,j-1}
                                     γ*_{i-1,j}
                                     γ*_{i,j-1} ]
    end for
  end for
end for
```

4. Distance Metrics for Phase Alignment

Phase Matching: Studies of human gesticulation [KvGvdH98, McN92, Ken80]) show that human gestures can be segmented into a sequence of discrete phases of different types, based on velocity and acceleration profile (see Table 1). For example, the "counting footsteps" motion, (Fig. 7) contains a repeated sequence of P,H,R phases, while the gesture for "Go down the driveway" in the "directions" sequence (Fig. 6) can be decomposed into phases P,S,H,R.

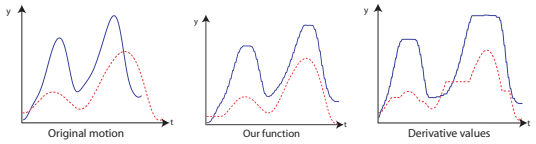


Figure 3: *In the first stage of our algorithm we align movement phases. For motions with significant amplitude differences, our distance function (center), which compares the signs of the first two derivatives over time, produces better results than a function using the values of the derivatives (right).*

Phase Type	Description	**v** and **a** profile
Stroke (S)	More force is exerted than in neighboring phases, indicated by acceleration or deceleration.	either $\mathbf{a} > 0$ or $\mathbf{a} < 0$
Hold (H)	Hand held still.	$\mathbf{a} = 0$, $\mathbf{v} = 0$
Preparation (P)	Non-stroke phase that departs from a resting position or joins two stroke phases.	$\mathbf{a} = 0$, $\mathbf{v} > 0$
Retraction (R)	Non-stroke phase that arrives at a resting position or switches into preparation (partial retraction).	$\mathbf{a} = 0$, $\mathbf{v} < 0$

Table 1: *Types of motion phases and their acceleration (**a**) and velocity (**v**) profiles (adapted from [KvGvdH98]). Positive velocity denotes movements outwards from the body, negative - towards the body.*

To match the corresponding phases in the two motions we use a distance function that evaluates the signs of the first and second derivatives over time for the two motion trajectories. The change in signs of derivatives reveals the changes in direction and velocity discontinuities that separate each hand motion phase. We chose to compare signs of derivatives rather than derivative values because, for motions with significant amplitude differences, the function based on derivative values produces non-uniform matching within phases, where many frames of one motion are aligned to a single frame of the other (see Figure 3).

To compute our function, we first subtract in each frame the shoulder position from the four hand markers in full-body animation. This isolates the movement of the arm and

the hand from the transition of the full-body motion. For example, when the character is walking and gesticulating, subtracting the shoulder position will extract the hand gesticulation and remove the transition effect from the hand markers. Next, we apply the time warping algorithm with distance metric defined as

$$\begin{aligned}\mathbf{D}(\mathbf{F}_1,\mathbf{F}_2) &= \sum_{i=1}^{4} \|\mathrm{sgn}(\dot{\mathbf{p}}_i) - \mathrm{sgn}(\dot{\mathbf{q}}_i)\|^2 \\ &\quad + \|\mathrm{sgn}(\ddot{\mathbf{p}}_i) - \mathrm{sgn}(\ddot{\mathbf{q}}_i)\|^2 \\ \mathbf{q}_i &= \mathbf{R}_y(\theta)\cdot \mathbf{p'}_i\end{aligned} \tag{6}$$

where $\mathbf{p}_i = [x_i, y_i, z_i]$ and $\mathbf{p'}_i = [x'_i, y'_i, z'_i]$ are the positions of the i'th marker in frames $\mathbf{F}_1$ and $\mathbf{F}_2$ respectively, $\dot{\mathbf{p}}_i = [\dot{x}_i(t), \dot{y}_i(t), \dot{z}_i(t)]^\mathsf{T}$ is the first derivative of marker positions over time and $\ddot{\mathbf{p}}_i$ is the second derivative. The sign function applied to a vector takes the sign of each vector element: $\mathrm{sgn}([x_i, y_i, z_i]^\mathsf{T}) = [\mathrm{sgn}(x_i), \mathrm{sgn}(y_i), \mathrm{sgn}(z_i)]^\mathsf{T}$.

Rotation about the vertical axis, $\mathbf{R}_y(\theta)$, locally aligns two motion fragments in space and can be computed with formula introduced in [KGP02], using small windows of neighboring frames around $\mathbf{F}_1$ and $\mathbf{F}_2$:

$$\theta = \arctan \frac{\sum_{j=1}^{n}(x_j z'_j - x'_j z_j) - \frac{1}{n}(\bar{x}\bar{z}' - \bar{x}'\bar{z})}{\sum_{j=1}^{n}(x_j x'_j + z_j z'_j) - \frac{1}{n}(\bar{x}\bar{x}' + \bar{z}\bar{z}')} \tag{7}$$

where barred terms are defined as $\bar{\alpha} = \sum_{j=1}^{n} \alpha_j$ and n is the total number of marker positions in a window. In our experiments we used windows with 20 marker positions (4 markers per frame for 5 frames).

Alternatively θ can be computed as an angle difference between the trunk orientations. While for the full-body sequence trunk orientation can be easily computed from the waist markers, in general it might not be available for the hand motion. However, if the hand data was recorded without significant trunk movement, like in our case, the trunk orientation is easy to obtain and constant throughout the motion. In practice, the method of computing θ from trunk angles is more robust but requires additional information. In our experiments, θ obtained from Equation 7 worked well for all motions except the Indian dancing, which involves a lot of variation and frequent orientation changes. For this motion we used the trunk angle method.

Matching frames within phases: The first stage of the algorithm aligns the corresponding phases of motions. Next, to refine the frame correspondence within the motion phases, we align the forearms of the hands in the two motions and minimize the angle differences between the palms of the hand in each frame (see Figure 5). Specifically, to compute the cost function between two frames $\mathbf{F_1}$ and $\mathbf{F_2}$, we first compute the forearm vectors $\mathbf{f}$ and $\mathbf{f'}$ and hand vectors

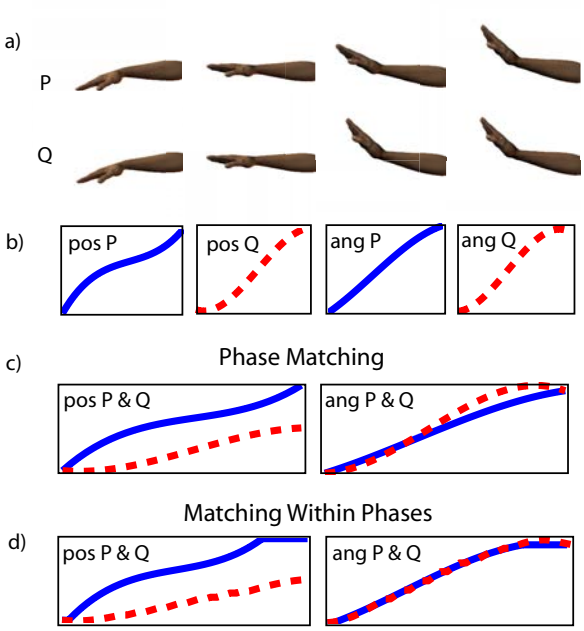


Figure 4: *After aligning phases of motion we refine frame correspondences within each phase. (a) Sequences of full-body motion (P) and hand motion (Q) belong to the same phase. (b) Graphs show wrist marker positions and hand angles for sequences P and Q. (c) Because all frames within a given phase have the same velocity and acceleration profile, during the first stage of our algorithm their alignment is arbitrary. (d) Second stage of the algorithm refines frame correspondences based on angle between forearm and palm of a hand.*

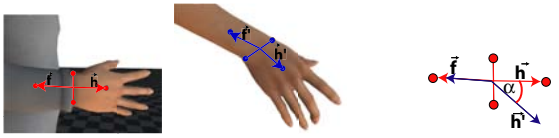
$\mathbf{h}$ and $\mathbf{h'}$ as

$$\begin{aligned}\mathbf{f} &= \mathbf{p}_3 - \frac{\mathbf{p}_1+\mathbf{p}_2}{2} \quad \mathbf{f'} = \mathbf{p'}_3 - \frac{\mathbf{p'}_1+\mathbf{p'}_2}{2} \\ \mathbf{h} &= \mathbf{p}_4 - \frac{\mathbf{p}_1+\mathbf{p}_2}{2} \quad \mathbf{h'} = \mathbf{p'}_4 - \frac{\mathbf{p'}_1+\mathbf{p'}_2}{2}\end{aligned} \tag{8}$$

where $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4$ and $\mathbf{p'}_1, \mathbf{p'}_2, \mathbf{p'}_3, \mathbf{p'}_4$ are the two wrist markers, the forearm marker and the hand marker for the full-body and hand motion respectively. Next we compute the rotation matrix $\mathbf{R}$ that aligns the two forearm vectors $\mathbf{f}$ and $\mathbf{f'}$ in space and compute the angle between two hand vectors:

$$\mathbf{D_2}(\mathbf{F_1},\mathbf{F_2}) = \arccos\left(\frac{\mathbf{h}\cdot\mathbf{R}\,\mathbf{h'})}{\|\mathbf{h}\|\|\mathbf{R}\,\mathbf{h'}\|}\right) \tag{9}$$

To adjust frame correspondence, we again use DTW along with the cost function $\mathbf{D_2}$. Here we use a narrower band to preserve phase correspondences while allowing adjustments to frame alignment within the phase. In our experiments the time warping band width was equal to 30% and 10% of the full-body motion length in the first and second stage respec-



(a) (b) (c)

Figure 5: $\mathbf{D_2}$ *denotes the angle between the hands in two motions. First we compute the forearm and hand vectors* $\vec{\mathbf{f}}$ *and* $\vec{\mathbf{h}}$ *for full-body motion (a) and* $\vec{\mathbf{f}}'$ *and* $\vec{\mathbf{h}}'$ *for hand motion (b). Next we align the forearm vectors* $\vec{\mathbf{f}}$ *and* $\vec{\mathbf{f}}'$ *and compute the distance function as the angle* α *between hand vectors* $\vec{\mathbf{h}}$ *and* $\vec{\mathbf{h}}'$ *after the alignment (c).*

tively. If necessary, we recompute the user's input so the frame numbers that need to be matched correspond to frames in the aligned motion.

Merging and smoothing: The frame correspondence resulting from the previous two steps goes through a final smoothing pass before rendering. Recall, the time warping algorithm produces matching sequences which may align a frame from one motion to one or more frames from the other motion. For the final animation, whenever a single frame from the full-body motion is matched to multiple frames from the hand sequence, we choose the median position of each marker from the frames and use it in the matched sequence. Because this frame averaging process can produce small discontinuities in the hand motion, as a final step, we blend discontinuities over a window of frames to create a smooth animation.

5. Results

We test our algorithm on various motions with complex hand gesticulation, where synchronization between the hand and body motions is crucial. Although our results are best seen in video form, we show sample frames of the resulting animations in Figures 1,6 and 7. Examples include: a series of counting animations, where the hand keeps count of the foot steps, an animation of a person giving a complex sequence of directions, Indian dancing motion where elaborate hand gestures (mudras) are used to tell a story and a more light-hearted animation of a charade for 'peeling a banana'. Each sequence contains multiple examples of aligned gesture phases (see Table 2).

All motions were captured with Vicon optical capture equipment (www.vicon.com). While the full-body animation was recorded without space restrictions, the hand motion was captured in a constrained area, roughly a 20" cube, where the arm's movement was severely limited. In all test scenarios our algorithm produced good results without the need for additional user input. For Indian dancing motion

Motion Sequence	Length (sec)	Number of Gestures	Number of Phases	Matching Points
Directions	18	6	16	0
Counting	16	8	20	0
Dance	18	34	52	3
Charade	9	4	6	0

Table 2: *In our experiments we used 4 motion sequences with varying number of gestures, from a simple charade to a complex Indian dance. All motions were aligned without user input, except for the Indian dance, where we specified 3 matching points. Numbers of gestures and phases are reported collectively for both hands.*

Motion Sequence	Number of Frames	Matching Time (sec.)
Directions	569	10.95
Counting	503	8.72
Dance	560	10.81
Charade	280	2.75

Table 3: *Running time of our algorithm using Matlab implementation.*

we specified 3 matching points to further increase the accuracy of the alignment. In the obtained motions the hand gestures are correctly synchronized with corresponding full-body movement. In the accompanying video we compare the resulting animations to the original human performer's moves during the motion capture recording. We also show an example of how user-specified constraints can be employed to achieve special effects, such as counting every left step.

6. Conclusion

Our straightforward but effective technique for aligning hand and full body motion exploits characteristics of human gestures in order to properly align motions with large amplitude differences that were captured separately, and at different resolutions. Thanks to its simple design, the proposed algorithm is efficient (see Table 3) and easy to implement. We have shown the power of our technique on motions with complex gesticulation and obtained correctly synchronized, natural-looking animation results.

There are several classes of motions for which our solution will likely fail. While our algorithm found appropriate matches for many free space gestures, gesticulation which involves specific spatial cues would require special consideration. For example, generating a motion with a constraint which ensures that a finger makes contact when touching the tip of one's nose lies outside the scope of the proposed solution. Additionally, motions which rely heavily on dynamics may not be amenable to the splicing described here. An example of such a motion may include shaking hands energetically, where the movement of the arms directly dictates the

Figure 6: *Example of directive gestures.*

Figure 7: *Counting example shows good synchronization of hand and body movement.*

motion of the hands. Finally, animations generated using our approach require that the motion be reasonably performed in a limited workspace for the hand capture. This is not the case in certain motions, like performing a gymnastic routine.

Ours is the first solution to a relevant and time-consuming process in production animation and should lead to other, more sophisticated automatic splicing approaches. Each of the classes of motion mentioned above introduce new, unique challenges and provide interesting directions for future work. We have limited our scope to the splicing of hand and full-body motions but many related problems arise when considering the splicing of multiple motion sources in a more general sense. We hope that our work will spur investigations in other areas where automatic motion alignment of separately captured motion performances is desired.

7. Acknowledgments

The work in this paper was partially supported by NSF under contract CCF-0429983. The authors would like to thank Nancy Pollard, Jessica Hodgins and Justin Macey for their assistance in collecting the motion capture data and the anonymous reviewers for helping to greatly improve the clarity of this paper. We would also like to thank Intel Corp., Microsoft Corp., ATI Corp. and Alias Corp. for their support through equipment and software grants.

References

[AF02] ARIKAN O., FORSYTH D. A.: Synthesizing constrained motions from examples. *ACM Transactions on Graphics 21*, 3 (July 2002), 483–490.

[BH00] BRAND M., HERTZMANN A.: Style machines. *Proceedings of SIGGRAPH 2000* (July 2000), 183–192.

[CH05] CHAI J., HODGINS J. K.: Performance animation from low-dimensional control signals. *ACM Transactions on Graphics 24*, 3 (Aug. 2005), 686–696.

[DYP03] DONTCHEVA M., YNGVE G., POPOVIĆ Z.: Layered acting for character animation. *ACM Transactions on Graphics 22*, 3 (2003), 409–416.

[FF05] FORBES K., FIUME E.: An efficient search algorithm for motion data using weighted pca. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2005), ACM Press, pp. 67–76.

[Gle97] GLEICHER M.: Motion editing with spacetime constraints. In *Proceedings of 1997 Symposium on 3D Graphics* (Apr. 1997), ACM SIGGRAPH, pp. 139–148.

[GMHP04] GROCHOW K., MARTIN S. L., HERTZMANN A., POPOVIĆ Z.: Style-based inverse kinematics. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 522–531.

[GSKJ03] GLEICHER M., SHIN H. J., KOVAR L., JEPSEN A.: Snap-together motion: Assembling run-time animation. *ACM Transactions on Graphics 22*, 3 (July 2003), 702–702.

[HGP04] HSU E., GENTRY S., POPOVIĆ J.: Example-based control of human motion. In *2004 ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (July 2004), pp. 69–77.

[HPP05] HSU E., PULLI K., POPOVIĆ J.: Style translation for human motion. *ACM Transactions on Graphics 24*, 3 (Aug. 2005), 1082–1089.

[IF04] IKEMOTO L., FORSYTH D. A.: Enriching a motion collection by transplanting limbs. In *2004 ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (July 2004), pp. 99–108.

[Ken80] KENDON A.: Gesticulation and speech: Two aspects of the process of utterance. In *Nonverbal Communication and Language* (1980), pp. 207–226.

[KG03] KOVAR L., GLEICHER M.: Flexible automatic motion blending with registration curves. In *2003 ACM SIGGRAPH / Eurographics Symposium on Computer Animation* (Aug. 2003), pp. 214–224.

[KG04] KOVAR L., GLEICHER M.: Automated extraction and parameterization of motions in large data sets. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 559–568.

[KGP02] KOVAR L., GLEICHER M., PIGHIN F.: Motion graphs. *ACM Transactions on Graphics 21*, 3 (July 2002), 473–482.

[KR05] KEOGH E. J., RATANAMAHATANA C. A.: Exact indexing of dynamic time warping. *Knowledge and Information Systems 7*, 3 (2005), 358–386.

[KvGvdH98] KITA S., VAN GIJN I., VAN DER HULST H.: Movement phase in signs and co-speech gestures, and their transcriptions by human coders. In *Proceedings of the International Gesture Workshop on Gesture and Sign Language in Human-Computer Interaction* (London, UK, 1998), Springer-Verlag, pp. 23–35.

[LCR*02] LEE J., CHAI J., REITSMA P. S. A., HODGINS J. K., POLLARD N. S.: Interactive control of avatars animated with human motion data. *ACM Transactions on Graphics 21*, 3 (July 2002), 491–500.

[LS99] LEE J., SHIN S. Y.: A hierarchical approach to interactive motion editing for humanlike figures. In *Proceedings of SIGGRAPH '99* (Aug. 1999), ACM SIGGRAPH, pp. 39–48.

[McN92] MCNEILL D.: *Hand and Mind: What Gestures Reveal about Thought*. UCP, Chicago, 1992.

[PB00] PULLEN K., BREGLER C.: Animating by multi-level sampling. *Computer Animation 2000* (May 2000), 36–42.

[PW99] POPOVIĆ Z., WITKIN A.: Physically based motion transformation. In *Proceedings of SIGGRAPH 99* (Aug. 1999), ACM SIGGRAPH, pp. 11–20.

[RCB98] ROSE C., COHEN M., BODENHEIMER B.: Verbs and adverbs: Multidimensional motion interpolation. *IEEE Computer Graphics and Applications 18*, 5 (1998), 32–40.

[RGBC96] ROSE C., GUENTER B., BODENHEIMER B., COHEN M. F.: Efficient generation of motion transitions using spacetime constraints. In *Proceedings of SIGGRAPH '96* (Aug. 1996), ACM SIGGRAPH, pp. 147–154.

[SCF04] SHAPIRO A., CAO Y., FALOUTSOS P.: Stylistic motion decomposition. *2004 ACM SIGGRAPH / Eurographics Symposium on Computer Animation - Poster* (Aug. 2004).

[SHP04] SAFONOVA A., HODGINS J. K., POLLARD N. S.: Synthesizing physically realistic human motion in low-dimensional, behavior-specific spaces. *ACM Transactions on Graphics 23*, 3 (Aug. 2004), 514–521.

[SP05] SULEJMANPASIĆ A., POPOVIĆ J.: Adaptation of performed ballistic motion. *ACM Transactions on Graphics 24*, 1 (Jan. 2005), 165–179.

[UAT95] UNUMA M., ANJYO K., TAKEUCHI R.: Fourier principles for emotion-based human figure animation. In *Proceedings of SIGGRAPH '95* (Aug. 1995), ACM SIGGRAPH, pp. 91–96.

[WH97] WILEY D. J., HAHN J. K.: Interpolation synthesis of articulated figure motion. *IEEE Computer Graphics & Applications 17*, 6 (Nov.-Dec. 1997), 39–45.

[WP95] WITKIN A., POPOVIĆ Z.: Motion warping. In *Proceedings of SIGGRAPH 95* (Aug. 1995), ACM SIGGRAPH, pp. 105–108.