Combined Partial Motion Clips

Nadia Al-Ghreimil
nadiag@gwu.edu

James K. Hahn
hahn@gwu.edu

The George Washington University
School of Engineering and Applied Science
801 22nd Street
Washington, D.C. 20052, USA

ABSTRACT

We present a motion editing method for articulated figures using Combined Partial Motion Clips (CPMCs). CPMCs contain detailed motion information for some parts of the articulated figure. They can be used to edit base motions in such a way that the parts that are not defined in detail will still be affected thereby emulating the correlation that exists naturally between joint movements. This is achieved through the inclusion of equations in the CPMC that capture the effects of the detailed motion on other degrees of freedom of the articulated figure.

Keywords
Computer Graphics, Animation, Motion Editing.

1. INTRODUCTION

Animating articulated figures is a difficult task due to the large number of degrees of freedom (DOFs) that have to be controlled. It gets even more difficult if the figure is a human, because we, the human observers, quickly detect unnatural movements. Therefore, reducing the number of DOFs that one has to deal with will simplify the task of the animator. A way to do that is to take advantage of the correlation that exists between the joint angles.

Motion capture is an effective way to obtain natural looking motions. A drawback of motion capture is that it is hard to maintain specific constraints and therefore often needs editing. For example, assume that a reaching motion from a sitting actor is recorded that will later be incorporated into an animation. It is possible that the animated character will not reach the desired position exactly, i.e. there is a difference in target reaching location. This is a case that calls for editing. A simple displacement map or inverse kinematics can be used to solve the problem.

But what if the difference is due to something else? Assume that in the animation the character is standing instead of sitting. Will it be possible to use the recorded 'reaching while sitting' motion? If so, then how will the legs be affected now that the character is standing?

Notice also that 'reaching' is an action mainly involving the upper body. Is it then necessary to capture and store joint trajectories of the lower body or are the joint trajectories of the upper body sufficient? If so, then how can we make it possible that a partial clip can be merged into a base clip and still look natural, i.e. not simply masking out some DOFs and replacing them with the ones in the stored clip (e.g. [Per95], [Ros96]) but overlaying and merging them with the existing motion and affecting the lower body appropriately?

Those questions drive our research.

We try to find relationships between body parts that hold for all subjects. We are striving not for physical correctness, but for a natural look. Additionally the necessary computations should be kept simple, and storage space requirements should be kept low.

We introduce a new kind of motion clips: Combined Partial Motion Clips or CPMCs, and a method for their generation and usage. A CPMC is a clip that is a combination of several partial motion data sets that have something in common: an action performed in various base poses. *Partial motion* means that only some joint trajectories are defined; the ones that are active in the common action. Furthermore, the CPMC contains equations to compute other joint trajectories from the ones included in the partial motion data set. This makes it possible for a CPMC to be used to add an action that primarily involves a specific part of the

WSCG SHORT PAPERS proceedings
WSCG'2003, February 3-7, 2003, Plzen, Czech Republic.

Changing end-effector position	Displacement mapping and waveshaping – [Bru95] e.g. knocking at a different height Warping – [Wit95] e.g. hitting a tennis ball at a different height Spacetime – [Gle97]	Motion Space – [Wil97] e.g. reaching to different locations Motion Space – [Ros98] e.g. reaching to different heights
Retargetting	Spacetime – [Gle98] Hierarchical curve fitting and IK- [Lee99] Physically based transforms – [Pop99]	Motion spaces could be used although there is no explicit example
Locomotion Characteristics	Signal Processing – [Bru95] e.g. can exaggerate a walk Fourier Representation – [Unu95] e.g. change slider to control step length	Differencing – [Unu95] e.g. extract briskness from walk, then add to run Motion Space – [Guo96] e.g. varying step length Motion Space – [Wil97] e.g. walking at different slopes Motion Space – [Ros98] e.g. walking and turning
Emotion	Add noise – [Per95] e.g. to simulate nervousness Signal Processing [Bru95] e.g. increase high frequency bands to simulate nervousness	Differencing – [Ama96] e.g. extract angriness from a drinking action and applying it to a kicking action
Transition	Overlap ends, find correspondence point with min. difference in position, velocity, and acceleration, distribute error over interval, Fit a least squares cyclic B-spline approximation – [Ros96] e.g. cyclification of walk	Spacetime – [Ros96] e.g. to concatenate several motions Motion Space – [Guo96] e.g. transition from walk to run Fourier Representation – [Unu95] e.g. interpolate to transition from walk to run Warping – [Wit95]
	Single file	**Multiple files**

Table 1: Categorization of Motion Editing Methods

body to some existing base motion while allowing the uninvolved parts to be affected in a reasonable way.

Our work is divided into two parts: the generation of Combined Partial Motion Clips, and their usage.

2. PREVIOUS WORK

The notion of motion editing became widespread with the increased popularity of motion capturing. The methods that arose are often applied to motion captured data, but usually are applicable to motion data generated in other ways, e.g. keyframing.

There has been a great deal of past research in the area of motion editing leading to the emergence of many motion editing methods. To mention a few, there are those that were applied to the data in the frequency domain [Unu95, Bru95], motion warping [Wit95] and displacement mapping [Bru95], spacetime constraint methods [Wit88, Coh92, Liu94, Gle97, Gle98, Pop99, Lee99], motion space methods (i.e., methods that use collections of motion samples then interpolate between them to generate new motions) [Wil97, Guo96, Ros98], differencing [Unu95, Ama96], etc. One can categorize these methods along two dimensions. First, according to the number of motion clips involved (single vs. multiple). Second, according to the purpose of editing. See Table-1.

‘Combining actions’ can be considered as another purpose of motion editing. By nature it involves multiple files. Two complete motions (involving all DOFs) may be combined, like blending two walks [Bru95]. By definition all DOFs will be affected. On the other hand, when a partial motion is combined with a complete one, it is not clear which DOFs will be affected. Bruderlin and Williams allow blending a walk with drumming [Bru95], Rose et al use their spacetime method to blend a walk with a salute [Ros96], and Perlin layers shrugging of the shoulders over standing [Per95]. In all these approaches, the original arm motion is masked out and the partial motion is inserted. Other DOFs, like the lower body, will not be affected. This may not be a disadvantage in the case of a salute, or waving, but for more vigorous motions of some body parts the resulting motion will look unnatural if the other parts of the body will just remain unchanged.

To our knowledge, there are no general methods that allow a partial motion to be combined with another motion and still affect all DOFs. This is where our research fits in.

More recently, methods that allow the addition of style or the synthesis of long, non-repetitive, smooth motions from a set of pre-stored motions have become known. Such methods analyze motion samples and find appropriate places for transitions. The samples are stored in graphs or in state machines. Transitions from one state to another or from node to node are then governed by probability functions thereby producing a continuous stream of motion [Bra00, Ari02, Kov02a, Lee02, Li02, Pul02].

Of these recent methods, the work of Pullen and Bergler [Pul02] is an example of combining partial with complete motions. Their method uses frequency analysis and creates pyramids, then uses the middle bands to find correspondences between the base motion and the motion from which to extract the ‘texture’ which in turn is contained in the higher bands. It is different from our work in that the base is a partial motion, which could be the keyframes of the legs for a walking motion. Motion captured data is then used to 'texture' the defined DOFs, while undefined ones can be synthesized from the same data by simply copying selected bands. Which DOFs will be affected and how much is controlled explicitly by the user. We on the other hand, add a partial motion to a base motion that can induce changes in other DOFs not included explicitly in the partial motion being added.

In the gaming industry partial motion clips are used extensively (e.g., for karate kicks, boxing movements, loading weapons, throwing, etc.). Still, there are no methods to combine them properly. Mostly, the partial motion that is added in does not affect the remaining DOFs and therefore the resulting motion often looks jerky and unnatural. We think that our research could be helpful in this regard.

3. TERMINOLOGY

In our work, we converted the raw data resulting from motion capture into root position and joint angles. Joint angles are expressed as XYZ-Euler angles and stored as vectors, $\mathbf{v}_i = [x\ y\ z]^T$ for joint i. The position of the root is expressed as a relative displacement in the root's local coordinate system. It also is stored as a vector, $\mathbf{v_0} = [x\ y\ z]^T$.

A character's configuration or *pose* is defined by the values of all its DOFs at an instance in time. Accordingly, a *partial pose* defines only some DOFs. A *trajectory* $\mathbf{q_i(t)}$ is a function that defines the values of $\mathbf{v}_i$ as it changes over time. The values of a character's (partial) pose as it changes over time make up a (*partial*) *motion*. It could be represented as a function, a sequence of keyframes, a collection of trajectories, or any other representation. A motion can be defined completely or partially.

Completely, meaning that the keyframes describe the whole character's configuration, or that all trajectories are stored:

$$\mathbf{m}(t) = \{\mathbf{q_i}(t): i = 0,1,\ldots, \text{NumberOfJoints}\},$$
$$\text{where } t = 0..\text{NumberOfFrames}-1.$$

Partially, meaning that the keyframes only describe part of the character's configuration, or that only the prominent trajectories are stored. Finally, we will call a motion to which we want to add a partial motion a *base motion*.

4. METHOD

Our objective is to be able to edit a base motion to generate a variation of it by combining it with a partial motion clip. We would like to be able to use the same partial motion clip to generate the same kind of variation in a multitude of base motions. For that, the partial motion clip has to be equipped with some 'knowledge' about the different base motions and how other DOFs will be affected in each case. The partial clip should be as compact as possible, yet allow as large a number of DOFs to be affected as necessary to make the resulting combined motion look natural.

4.1 CPMC Generation

Here we will explain how to extract a partial motion from various motion-captured clips and how to turn it into a CPMC. See figure-1.

4.1.1 Data Collection and Manipulation

Motion data for the desired action (i.e., that causes the variation) needs to be collected. The data should include multiple samples where the action is performed during different base motions, e.g., reach while walking, reach while jumping, and reach while squatting. The base motions without the desired action also need to be collected. Samples should be taken from several subjects for a more complete analysis.

The collected motions need to undergo some pre-processing to make them ready for differencing. The main task is to align the motions based on some landmarks, e.g. heel-strikes, maximum arm extension, etc. Shifting, resampling, and trimming usually accomplish the task.

4.1.2 Differencing

Similar to the extraction of *emotion* [Ama96] or the extraction of *characteristics of a walk* [Unu95] we will extract an *additional movement of part of the body* using a simple vector difference of each corresponding trajectory at each frame from the base motions.

Let $\mathbf{m_{b1}}(t)$ define a base motion, and $\mathbf{m_{d1}}(t)$ define a motion of the same subject performing the desired action during the same base motion. We extract the desired motion $\mathbf{m_1}(t)$ by taking the difference. We repeat the same for all samples of the desired action and their corresponding base motions:

$$\mathbf{m_i}(t) = \mathbf{m_{di}}(t) - \mathbf{m_{bi}}(t) \quad \text{for } i=1..\text{NumberOfBases} \quad (1)$$

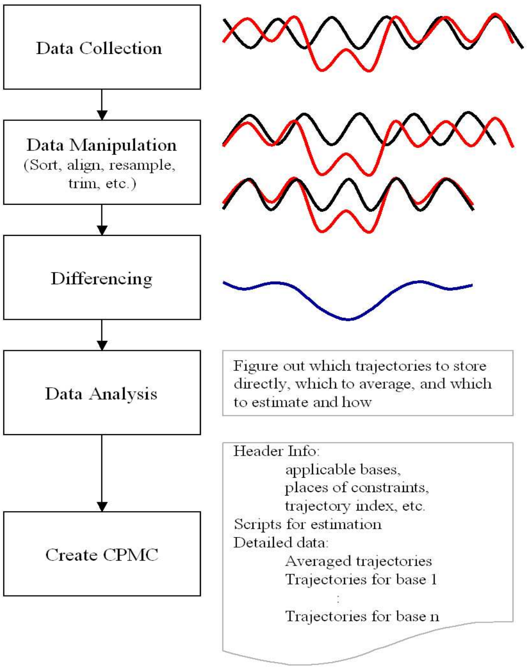

Figure 1: Generating a CPMC

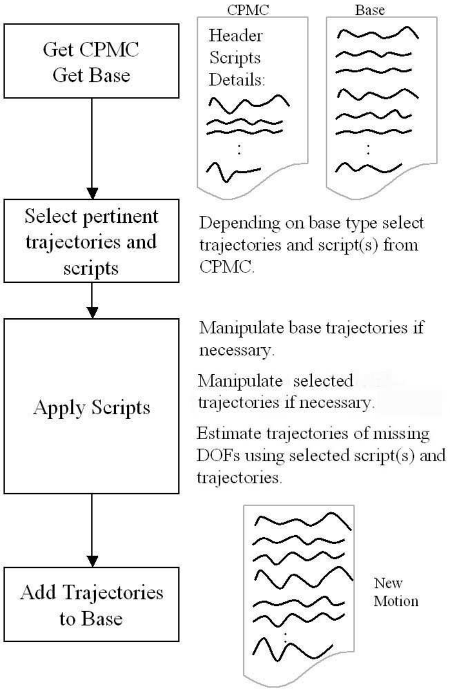

Figure 2. Using a CPMC

This will result in a collection of extracted desired actions, one for each base at least. The process is repeated for all subjects leading to more collections, one for each subject. At this stage all motions are still stored as complete motions.

4.1.3 Data Analysis

We want to store a minimal amount of data without losing the details of the motion that characterize it, i.e. we want to store the extracted action as a partial motion. For that, the extracted difference motions ($\mathbf{m_i}$) are carefully analyzed in order to find out which trajectories need to be stored as they are, which can be averaged (across the bases and/or subjects) and which ones can be recovered through a relationship with the stored ones.

The question we ask is: What are the similarities and differences

1. between all $\mathbf{m_i}$s of one subject?
2. between subjects for a specific $\mathbf{m_i}$, i.e. the ones extracted from a specific base motion?

To answer those questions one has to compare the joint trajectories and their derivatives. Trajectories that are nearly identical in all $\mathbf{m_i}$s of one subject are usually the ones that are the main constituents of the desired action being extracted, for example, the trajectories of the arm joints in a reaching motion.

When analyzing the original motion data it is often hard to see how similar the effects of a particular DOF are on another for the different base motions. Since we are not interested in the correlation between the joints in the base motion itself, but in the correlation between the DOFs involved in the partial motion and other DOFs independent of the base, we analyze the changes, i.e. the difference trajectories taken between the detailed motion and the bases, because this takes out the effect of the base motion and enables us to find more accurate relationships between the DOFs of the partial motion and the DOFs of the base motion. These relationships can then be used to approximate the trajectories that will not be stored.

Furthermore, depending on the action on hand, one may turn to the research done in the field of biomechanics for help in analyzing the data and obtaining relationships between the movements of the joints.

4.1.4 Clip Combination

The results of the analysis dictate the way the difference motions are combined. The trajectories that are nearly identical across the bases can be averaged and stored. Whether one uses the data from one subject or all subjects depends on how much variation there is between them. For trajectories that can be estimated from their relationship with stored trajectories, the scripts for their computation are stored. The remaining trajectories making up the partial motion need to be stored individually for each base.

This combination leads to a reduction in storage space requirements. Let the number of bases be **b**, the number of trajectories that constitute the extracted partial action be **n**<(NumberOfJoints+1), the number of averaged trajectories be **a** (i.e., the number of trajectories stored separately for each base = **n**-**a**), and the estimated ones be **e**. With the currently available editing methods, one would have to store (NumberOfJoints+1)***b** trajectories to be able to add the desired motion to another one and affect all DOFs. If only a partial motion is stored then only **n*****b** trajectories need to be stored. Since some trajectories can be averaged, a further reduction is possible, and the total number of trajectories stored is only:

$$\mathbf{a}+(\mathbf{n}-\mathbf{a})*\mathbf{b} \qquad (2)$$

Hence, the total reduction is equal to:

$$(\mathbf{e}+\mathbf{a})*\mathbf{b}-\mathbf{a} \qquad (3)$$

Current methods would only affect **n** trajectories for each base, our method allows the editing to affect **n**+**e** trajectories for each base.

All the trajectories selected for storage and the scripts need to be preceded by an appropriate header. The header should contain information about constraint positions if necessary, like heel strike locations, or maximum arm extension, etc. It also should state which bases this CPMC applies to, and contain an index of the trajectories and scripts, and pointers to them.

4.2 CPMC Usage

A CPMC is used by adding it to a base motion. See figure-2. The main information needed for a successful addition of a base motion and a CPMC is the type of base motion. Some base motions may need additional information, e.g., the cycle length and the time of the first right heel strike in a running base motion. Which data is needed exactly should be identifiable from the header information of the CPMC and the scripts for that particular base.

Once the type of base motion is known, the appropriate trajectories and scripts can be selected from the CPMC with the help of the indexes and pointers contained in the header. Some base motions may require some preprocessing, for example resampling. Also, the selected trajectories need to be matched to the base motion since the CPMC is created for base motions with specific parameters, in the case of running it may be step length, jump height, and cycle length. After that additional trajectories can be computed according to the scripts.

Since a CPMC is a result of a difference, the last step is to add all the computed and selected trajectories to their corresponding trajectories in the base motion using a simple vector addition at each frame to get the final result.

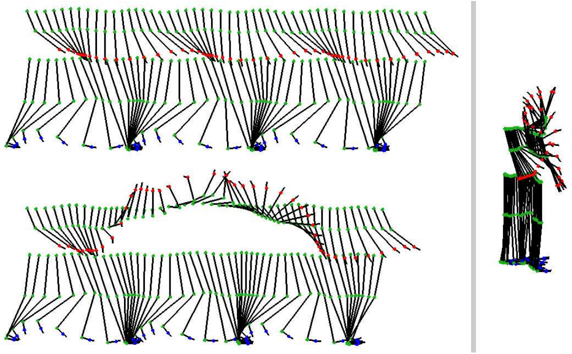

Figure 3. Multiple exposures of the right arm and leg of a walk (left top), walk&throw (left bottom). and their difference (right)

5. EXPERIMENTS

We tested our approach on the specific example of throwing. We successfully extracted the partial throwing motion performed during three base motions: walking, standing, and sitting. We then created a CPMC for throwing, which we used to edit base motions from several subjects.

5.1 Generating a CPMC for 'throwing'

The overhand throwing motion of 6 subjects (3 males and 3 females, with different heights) was obtained via optical motion capture. The subjects had 26 light reflecting markers attached to their body (5 on each leg, 6 on each arm, 1 on top of each shoulder, 1 on the neck, and 1 on the back). The action was performed using a lightweight sponge ball during walking, during standing, and during sitting. The subjects also performed just walking, standing, and sitting. Each motion was captured several times until at least 5 acceptable recordings were obtained. The data was converted to root position and joint angles as explained in section 3.

The motions of each subject were prepared for differencing. 'Walk' $\mathbf{m_w}(t)$ and 'Walk&Throw' $\mathbf{m_{wt}}(t)$ motions had to be aligned properly. It is usually the case that one slows down while throwing. Therefore, we resample part of the walking motion $\mathbf{m_w}(t)$ starting at a heel strike that matches the one just before the initiation of throwing in $\mathbf{m_{wt}}(t)$ and ending at a heel strike just after the throw when normal walking is resumed so that its duration is matched with that in $\mathbf{m_{wt}}(t)$, the *throw-period*. In a right-handed throw the throw-period may start as early as a left heel strike before the right arm starts the wind-up phase. It ends a few steps later at a right or left heel strike, differing from one person to another, after the arm is back to its normal position, i.e. after the follow-through phase, when normal walking can be resumed (see figure-3).

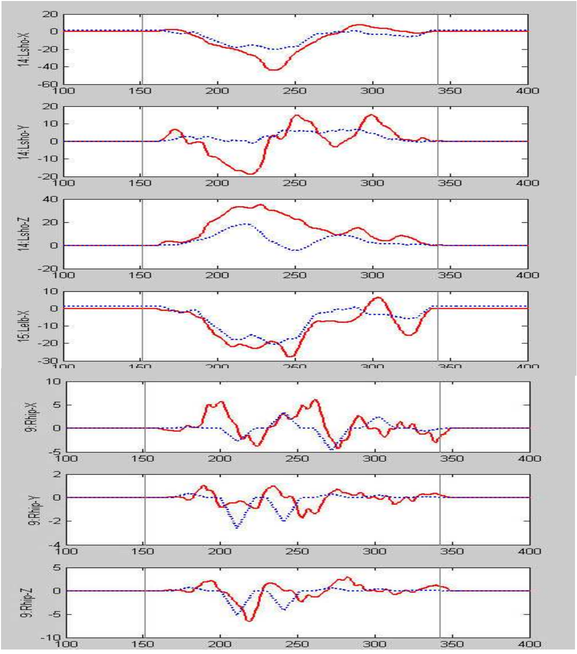

Figure 4. Comparison of difference trajectories: actual (solid red) and estimated (dotted blue).

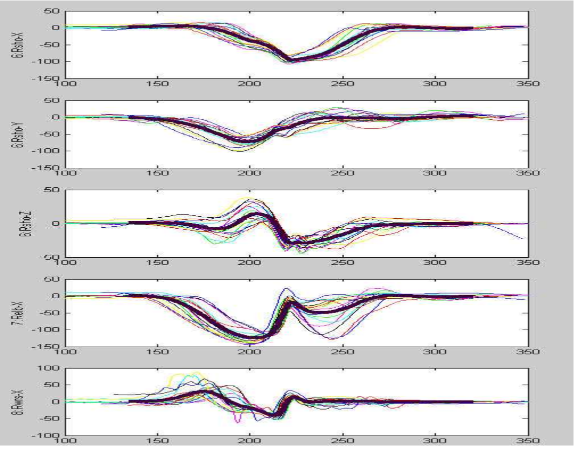

Figure 5. Throwing-arm difference trajectories (thin colored) and their average (thick black).

There is no need to match directions explicitly because the position of the articulated figure is expressed as a displacement in the local coordinate system of the root. Neither does one need to match step lengths because the change in step length will be computed in view of the fact that $\mathbf{v_0}$ is included in the differencing process.

Since we have three base motions we will have several samples of $\mathbf{m_1}(t)$, $\mathbf{m_2}(t)$, and $\mathbf{m_3}(t)$ for each subject as a result of the differencing process. These samples were then analyzed comprehensively.

In the field of biomechanics, a lot of research is devoted to throwing. The various kinds of throwing are explored, e.g. baseball pitching, football passing, javelin, etc. The main purpose of the research though is to explain the mechanics of throwing and prevention of injuries [Fle96]. We did not find anything specific about the coordination of the upper body and the lower body during throwing other than timing relationships [Ell88].

By looking at joint trajectories and their derivatives, we noticed that the throwing-arm motion was almost identical in $\mathbf{m_1}$, $\mathbf{m_2}$, and $\mathbf{m_3}$ for a specific subject, but would differ across subjects because of their distinct throwing style. Nevertheless, we found that the changes in the non-throwing arm for example would be similar across subjects and even across bases, and could therefore be computed using the same equations.

More specifically, we found that the y-rotation of the trunk affects the swinging of the non-throwing arm, e.g. the x-rotation of the shoulder. The non-throwing arm is also affected by the motion of the shoulder of the throwing arm. And the changes in the elbow and wrist of the non-throwing arm are merely scaled versions of the changes in the shoulder.

As for the legs, we only need to compute their changes for the case of walking. For sitting and standing, one can simply apply inverse kinematics to keep the feet steady on the ground since the root position and orientation and therefore the hip positions are known.

From the comparison of the leg joint trajectories in $\mathbf{m_w}(t)$ with those in $\mathbf{m_{wt}}(t)$ it is apparent that there is only a slight increase/decrease in the hip rotation at the peaks/valleys, which coincide with right and left heel strikes. The changes in the hips are a way to compensate for the changes in the body's trajectory and the amount of change is related to the change in the position of the root. Therefore, a simple displacement map can be constructed. Changes in knees and ankles are similar (see figure 4).

From this analysis we concluded that in addition to cycle length and the equations or scripts for computing the non-throwing arm and the legs, we only need to store the trajectories of the trunk (root, and back), the position, and the throwing arm (shoulder, elbow, and wrist), i.e., $\mathbf{n}=6$.

Since the changes of the trajectories of the throwing arm ($\mathbf{a}=3$) are very similar during all three base motions ($\mathbf{b}=3$) we only store their average (see figure 5). On the other hand, the trajectories of the root, back, and position are too different and need to be stored individually for walking, standing, and sitting. Note that if a complete motion had been used then 15*3 trajectories would have been stored. But using the CPMC the total number of trajectories stored will only be 12 (equation 2) leading to a total reduction of 33 trajectories (equation 3). Furthermore, only 6 trajectories will be used at a time, but the total

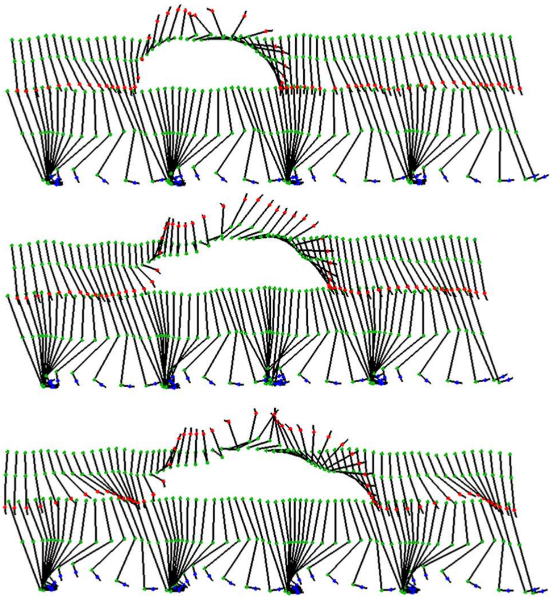

Figure 6. Multiple exposures of the right arm and leg of 3 walk&throws: original of subject b (top), generated for subject b from CPMC (middle), original of subject a whose data is used in the CPMC (bottom).

number of trajectories that will be affected in the resulting motion will be 15, since we can estimate 9 trajectories (**e**=9).

We used the data of one subject only in the final CPMC. If a different style of throwing is desired, another CPMC can be created by using the motion data from another subject with that specific style. The same equations can be used since we used the data of all subjects for analysis.

5.2 Using a CPMC for ‘throwing’

This CPMC was then used to add a throwing motion to other subject’s walking, standing, and sitting base motions. The base walking motions differed in their cycle length and style.

The simplest case is sitting. We merely add the averaged throwing-arm data and the root, back, and position data specific to sitting. The non-throwing arm and the legs can be fixed in place using inverse kinematics.

The next case is standing. This time the non-throwing arm will be computed using the equations that resulted from the analysis and that are stored within the CPMC. The resulting data, the averaged throwing-arm data and the root, back and position data specific to standing are added. The legs, like in sitting, can be fixed with inverse kinematics.

The last case is walking; here a little more processing is needed. First, we find a period with the appropriate number of heel strikes, matching the ones in the throw-period of the CPMC. And resample it to slow it down to about 80-90% of the original speed. The amount of slow-down is related to how vigorous the throw is, which we measure by the peak velocity of the elbow and shoulder.

Now, the averaged throwing-arm data and the root, back, and position data specific to walking has to be resampled to match the length of the slowed-down throw-period. Then, the data for the non-throwing arm is computed as well as for the legs. Finally, all the data can be added to the corresponding data in the base motion (see figure 6).

6. RESULTS

The motions resulting from our editing method, as tested in the experiment above, are very encouraging. We did not put any limits on the joint angles; which may have improved the results.

A problem that we encountered is the slipping of the feet, but this is a minor problem that all motion editing methods suffer from with the exception of methods that explicitly put constraints on the foot locations. The problem is usually corrected during post-processing. One could use such a method as that of Kovar et al [Kov02b] for the walking case. During standing and sitting we used an inverse kinematics method that has been developed at the University of Pennsylvania in the Center for Human Modeling and Simulation called IKAN.

We allow the user to specify a resampling rate during walking to be applied instead of the default value. Other variables allow the user to exaggerate the throwing motion.

We briefly tried our method for a different kind of throw. We asked our subjects to throw the ball as if they were tossing an empty soda can into a trashcan, again while walking, standing, and sitting. We created a CPMC and were able to use the same scripts as those for throwing. This indicates to us that for similar partial motions the relationship to other body parts is similar. But we think that for radically different actions, e.g., for boxing, the relationships are probably different. Still, since human motion is highly correlated one may be able to find very general relationships between the different DOFs of the articulated figure.

7. CONCLUSION

We presented a method for editing motions in such a way that the effects of some DOFs on others is taken into consideration. For that purpose we introduced CPMCs. The method describes how to combine a partial motion with a base motion while allowing the added part to affect DOFs of the base motion not included in the partial motion.

The CPMCs we talked about apply to different base motions. One could generalize the idea by allowing multidimensional variations. For example, our throwing CPMC which applies to 3 base motions could be extended to apply to different masses of objects being thrown. Another dimension could be the distance thrown. For that, one would need to collect samples along each dimension. Then for each dimension eliminate the effect of the variable by subtracting it from a reference motion, for example the one with the smallest mass and shortest distance.

Collaboration with other disciplines, statistics and biomechanics, would help in the analysis of the data and in the identification of the relationships between the DOFs, which is the heart of our method.

8. ACKNOWLEDGMENTS

We would like to thank Dr. Steven Stanhope and his staff at the biomechanics laboratory at the Warren Grant Magnuson Clinical Center at the National Institutes of Health (NIH) for allowing us to use their laboratory space and motion capture equipment. Special thanks to the people who were our subjects.

9. REFERENCES

[Ama96] Amaya, K., A. Bruderlin, and T. Calvert. “Emotion from Motion.” In Graphics Interface 1996, pp.222-229.

[Ari02] Arikan, O. and D. A. Forsyth. “ Interactive Motion Generation from Examples.” In Proceedings of SIGGRAPH 2002, Aug. 2002, pp.483-490.

[Bra00] Brand, M., and A. Hertzmann. “Style Machines.” In Proceedings of SIGGRAPH 2000, Aug. 2000, pp-183-182.

[Bru95] Bruderlin, A., and L. Williams. “Motion Signal Processing.” In Proceedings of SIGGRAPH 1995, Aug.1995, pp.97-104.

[Coh92] Cohen, M. “Interactive Spacetime Control for Animation.” In Proceedings of SIGGRAPH 1992, 1992, pp.293-302.

[Ell88] Elliot,B., J. Robert Grove, and B. Gibson. “Timing of the Lower Limb Drive and Throwing Limb Movement in Baseball Pitching.” International Journal of Sport Biomechanics, 4, 1988, pp.59-67.

[Fle96] Fleisig, G.S., S.W. Barrentine, R.F. Escamilla, J.R. Andrews. “Biomechanics of overhand throwing with implications for injuries.” Sports Medicine Review. 21(6), June 1996, pp.421-437.

[Gle97] Gleicher, M. “Motion Editing with Spacetime Constraints.” In Interactive 3D Graphics 1997, pp.139-148.

[Gle98] Gleicher, M. “Retargetting Motion to New Characters.” In Proceedings of SIGGRAPH 1998, Aug.1998, pp.33-42.

[Guo96] Guo, S., and J. Robergé. “A High-Level Control Mechanism for Human Locomotion Based on Parametric Frame Space Interpolation.” In Computer Animation and Simulation, Eurographics Workshop, Springer-Verlag, 1996, pp.95-107.

[Kov02a] Kovar, L., M. Gleicher, and F. Pighin. “Motion Graphs.” In Proceedings of SIGGRAPH 2002, Aug. 2002, pp.473-482.

[Kov02b] Kovar, L., M.Gleicher, and J. Schreiner. “Footskate Cleanup for Motion Capture Editing.” In proceedings of the ACM SIGGRAPH symposium on Computer animation, July 2002, pp.97-104.

[Lee02] Lee, J., J. Chai, P.S.A. Reitsma, J. K. Hodgins, and N. S. Pollard. “Interactive Control of Avatars Animated with Human Motion Data.” In Proceedings of SIGGRAPH 2002, Aug. 2002, pp.491-500.

[Lee99] Lee, J., and S.Y. Shin. “A Hierarchical Approach to Interactive Motion Editing for Human-like Figures.” In Proceedings of SIGGRAPH 1999, 1999, pp.39-48.

[Li02] Li, Y., T. Wang, and H.Y. Shum. “Motion Texture: A Two-Level Statistical Model for Character Motion Synthesis.” In Proceedings of SIGGRAPH 2002, Aug. 2002, pp.465-472.

[Liu94] Liu, Z., S. Gortler, and M. Cohen. “Hierarchical Spacetime Control.” In Proceedings of SIGGRAPH 1994, 1994, pp.35-42.

[Per95] Perlin, K. “Real-time Responsive Animation with Personality.” In IEEE Transactions on Visualization and Computer Graphics, 1(1), March 1995, pp.5-15.

[Pop99] Popovic, Z., and A Witkin. “Physically Based Motion Transformation.” In Proceedings of SIGGRAPH 1999, Aug.1999, pp.11-20.

[Pul02] Pullen, K., and C. Bergler. “Motion Capture Assisted Animation: Texturing and Synthesis.” In Proceedings of SIGGRAPH 2002, Aug. 2002, pp.501-508.

[Ros96] Rose, C., B.Guenter, B. Bodenheimer, and M. Cohen. “Efficient Generation of Motion Transitions using Spacetime Constraints.” In Proceedings of SIGGRAPH 1996, Aug.1996, pp.147-154.

[Ros98] Rose, C., M. Cohen, B. Bodenheimer. ”Verbs and Adverbs: Multidimensional Motion Interpolation.” In IEEE Computer Graphics and Applications, 18(5), Sept/Oct 1998, pp.32-40.

[Unu95] Unuma, M., K. Anjyo, and R. Takeuchi. “Fourier Principles for Emotion-based Human Figure Animation.” In Proceedings of SIGGRAPH 1995, Aug.1995, pp.91-96.

[Wil97] Wiley, D. J., and J. Hahn. “Interpolation Synthesis of Articulated Figure Motion.” In IEEE Computer Graphics and Applications, 17(6), Nov/Dec 1997, pp.156-160

[Wit88] Witkin, A., and M. Kass. “Spacetime Constraints.” In Proceedings of SIGGRAPH 1988, 1988, pp.159-168.

[Wit95] Witkin A., and Z. Popovic. “Motion Warping.” In Proceedings of SIGGRAPH 1995, Aug.1995, pp.105-108.