Articulated Video Sprites

C. Vanaken and M. Gerrits and P. Bekaert

Hasselt University
Expertise Centre for Digital Media
transnationale Universiteit Limburg
Wetenschapspark 2, BE-3590 Diepenbeek (Belgium)
cedric.vanaken,mark.gerrits,philippe.bekaert@uhasselt.be

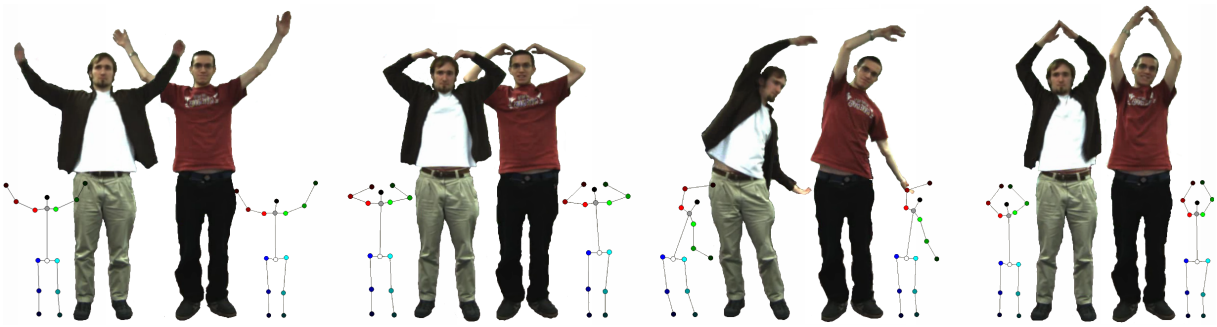


Figure 1: *We acquire a clip of the person on the right, consisting of arbitrary movements. Next, 2D skeletons are extracted semi-automatically, providing a high level description of the recorded motion. In particular, we can use the skeletons to match frames to a user-specified 2D motion. In this example, we used the extracted skeletons of the left person to animate the right person.*

Abstract

In this paper, we present an extension to video sprites for articulated characters. Central to our technique is a matching cost which works on high-level 2D skeletal representations of the characters, instead of their visual appearance. Through a combination of different heuristics, we are able to animate a character according to a new sequence of target skeletons. This way we achieve an accurate matching and a hitherto unseen level of control over the video sprite.

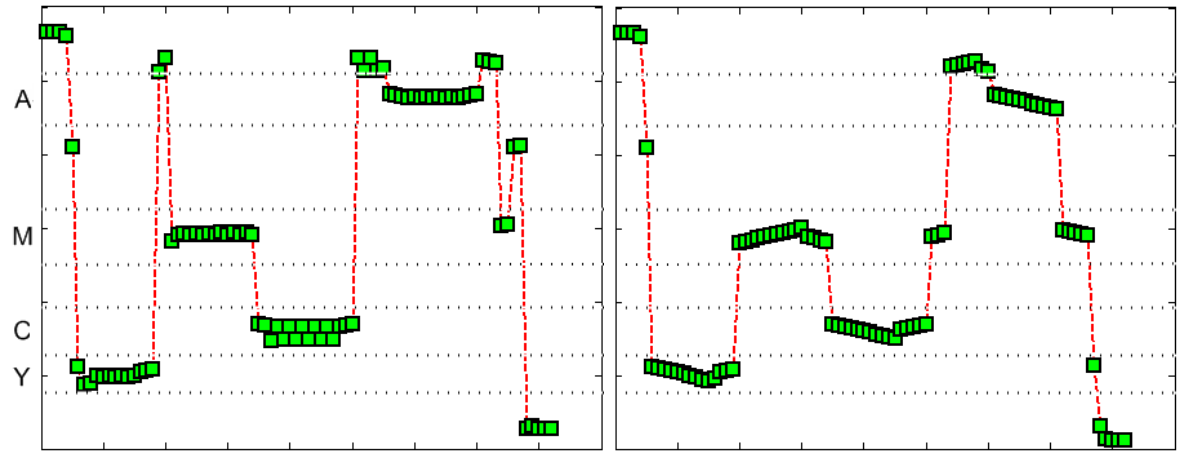
Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism— Animation; I.4.9 [Image Processing And Computer Vision]: Applications

1. Introduction

We present a new approach to video sprites that allows for animation of articulated characters, such as animals or humans. Video sprites [SE02] are used to animate characters in a data-driven way by rearranging existing video frames, while maintaining smooth temporal coherence and the natural appearance present in the captured footage. The key to our technique is a matching algorithm that focuses on high-level 2D skeleton models instead of the character's visual appearance.

Traditional video sprites focus on simple characters without articulation (e.g. fish) or cases where the effects of articulation are negligible (e.g. flies or hamsters). This limits the usability of the video sprites technique and does not allow for a high level of control over the animation. Often it is necessary to control the animation in more detail (to the level of character poses), for instance to infuse emotion into a character's performance.

Articulated characters possess an underlying structure in the form of skeletons. We exploit this by using skeletal rep-



(a) Without penalties (b) With penalties

Figure 2: *Results of the matching algorithm. The nodes on the graphs indicate which source frames (vertical axis) are chosen as best matches for the target animation frames (horizontal axis). As can be seen in the left figure, without penalty costs freeze frames and oscillations occur.*

resentations in our algorithm, instead of the visual appearance of the subject. The desired animation is defined as a sequence of skeletons. These can be acquired in a number of different ways, such as by motion capture or provided by an animator. These target skeletons are then matched with skeletons previously extracted from the source footage. The target skeletons only function as a guide for the new animation and do not force the input frames to match them exactly. This way the animation will achieve the desired effect without sacrificing the natural movement and appearance of the filmed subject.

Our approach differs from most contemporary methods [CTMS03] to animate articulated characters, in that we start from two-dimensional footage instead of data acquired from multiple viewpoints. As such, a less restrictive recording setup is required for our method and we can even incorporate existing footage which was not originally shot to be reanimated.

2. Algorithm

Our method starts with a preprocessing step, in which sufficient video footage is collected, from which 2D skeletons are semi-automatically extracted. Next, we find the best matches, incorporating several heuristics, between these skeletons and the target sequence of new skeletons. Finally, our results are refined by taking into account blocks of distinct motions.

Throughout this paper, we will illustrate our techniques on a specific example. We start from a large source video in which the subject character performs several movements, amongst them the arm gestures for the letters Y, M, C and A. The target animation to which we want to match our source video consists of another person performing the famous YMCA routine. The results of our algorithm are shown in Figure 1.

2.1. Preprocessing

Similar to previous techniques [SSSE00, SE02], we start by amassing a sufficient amount of video material of the subject that we wish to animate. This footage can be acquired in a studio, on location, or from already existing material. Because we will be reusing existing frames, every posture of the subject in the target animation must approximately appear in the source footage.

The 2D skeletons are semi-automatically extracted from the source footage with user-assisted optical flow tracking. The user indicates all skeleton joint positions in the first frame. These are then tracked across the entire video by an optical flow algorithm. We chose to use Lucas & Kanade's technique [LK81]. Occlusions are handled by having the user indicate the problematic frames. The joint positions in these frames are then interpolated based on its positions in several frames before and after the occlusion. During this step, the subject can also be optionally segmented from the footage.

The target animation is provided as a sequence of target skeletons. They can be produced by an animator, provided through motion capture, or extracted from video footage the same way as the source skeletons. The only requirement is that both sequences feature the same skeletal structure. For humans, we chose a simple skeleton, consisting of 14 connected limbs, as can be seen in Figure 1.

2.2. Skeleton matching

To find the correct order of source frames for the new animation, we match the skeletons extracted from the source material to the provided target skeletons. At the same time we also ensure that the chosen skeletons follow each other in a smooth temporal manner by forcing consecutively chosen skeletons to have small matching costs. For our matching cost, we do not compare absolute positions of the skeleton joints, as these are dependant of the scene and body type of the actors or animation models. Instead, we compare the angles and ratios between adjacent limbs. These codify the 2D posture of the skeleton sufficiently and, through the ratios, implicitly include some 3D information.

To avoid perceptually disturbing artefacts, we penalize sequences where the same frame occurs many times in a row (freeze) or where it oscillates between nearby frames (stutter). Using dynamic programming, we combine these costs to find an optimal new frame order. The efficiency of the introduced penalties is shown in Figure 2. The nodes on the graphs indicate which source frames (vertical axis) are chosen as best matches for the target animation frames (horizontal axis). As can be seen in Figure 2(a), the absence of penalties results in a fluctuating and rather unstructured frame order, while fairly smooth results are obtained by incorporating the penalties (Figure 2(b)). This whole process happens fully automatically.

2.3. Matching Refinement

Motion of articulated characters, such as humans, can often be subdivided into a chain of distinct movements. We automatically detect these atomic movements by analyzing the sequential matching costs in the skeleton sequences. Their boundaries are detected as local maxima of this function.

After a global pass of our matching algorithm, we refine our results by restricting the matching process to smoothly stay inside the bounds of these movements. By imposing this restriction, we avoid mixing several of these distinct motions in the resulting animation, which leads to temporal artefacts. Furthermore, we introduce a new penalty to inhibit disproportionally large jumps inside these single motions. Finally, an averaging window is applied inside the distinct movements to further improve local temporal smoothness. While in Figure 2(b), the letter A in the new animation was still composed from frames from the letters M and A, the letter A has now become more consistent and coherent consisting of only A frames as can be seen in Figure 3.

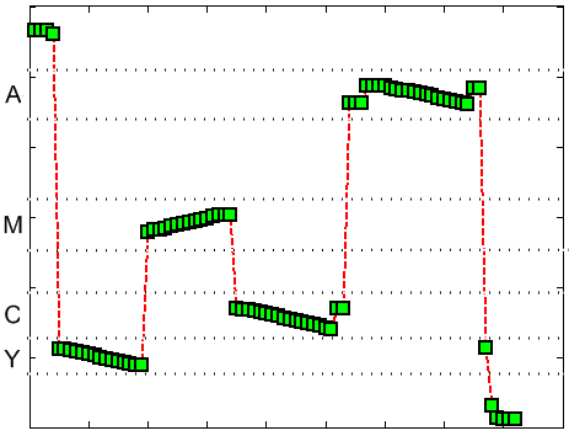


Figure 3: *Final frame selection after matching refinement.*

3. Results

As shown in Figure 1, our algorithm correctly matches different poses even though the visual appearance differs greatly and the skeletons don't match exactly. Furthermore, a large variety of poses were present in the original source footage. The relevant movements and poses were extracted from this data, while the other poses were succesfully filtered out. The included video shows the full animation from which the examples in the figure were extracted.

4. Conclusion and Future Work

In this paper, we presented a technique to animate articulated video sprites. We achieved this by automatically matching their skeletal representations instead of their visual appearance, granting us accurate matching and a high level of control over the animation.

In the future, we hope to extend the work presented here by inter- and extrapolating characters for poses missing from the original source data. We also plan to allow the user to manually steer the matching process and to introduce a hierarchical skeleton model to allow for an adaptable level of detail in the skeletons, for instance by switching to a detailed skeleton of a person's hand in a close-up.

References

[CTMS03] CARRANZA J., THEOBALT C., MAGNOR M. A., SEIDEL H.-P.: Free-viewpoint video of human actors. *ACM Trans. Graph. 22*, 3 (2003), 569–577.

[LK81] LUCAS B. D., KANADE T.: An iterative image registration technique with an application to stereo vision (ijcai). In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81)* (April 1981), pp. 674–679.

[SE02] SCHÖDL A., ESSA I. A.: Controlled animation of video sprites. In *SCA '02* (2002), ACM Press, pp. 121–127.

[SSSE00] SCHÖDL A., SZELISKI R., SALESIN D. H., ESSA I.: Video textures. In *Siggraph 2000* (2000), pp. 489–498.