Computational Visual Media
https://doi.org/10.1007/s41095-018-0115-y Vol. 4, No. 3, September 2018, 197–208

Research Article

Dance to the beat: Synchronizing motion to audio

Rachele Bellini[1] (✉), **Yanir Kleiman**[1], **and Daniel Cohen-Or**[1]

Abstract In this paper we introduce a video post-processing method that enhances the rhythm of a dancing performance, in the sense that the dancing movements are more in time to the beat of the music. The dancing performance as observed in a video is analyzed and segmented into motion intervals delimited by motion beats. We present an image-space method to extract the motion beats of a video by detecting frames at which there is a significant change in direction or motion stops. The motion beats are then synchronized with the music beats such that as many beats as possible are matched with as little as possible time-warping distortion to the video. We show two applications for this cross-media synchronization: one where a given dance performance is enhanced to be better synchronized with its original music, and one where a given dance video is automatically adapted to be synchronized with different music.

Keywords video processing; synchronization; motion segmentation; video analysis

1 Introduction

Dancing is a universal phenomenon, which crosses cultures, gender, and age. Dancing is even observed in some animals in the wild. We all appreciate and enjoy good dancing; however, an interesting question is what makes a dancing performance look good, and can we enhance a dancing performance as observed in a video? One critical aspect of a good dance performance is that the movement is in time, i.e., with good synchronization between the dancing movement and the music [1]. In this paper, we introduce a video post-processing technique that enhances the rhythm of a dancing performance so that the dancing movements are in better time to the music.

Studies dealing with dance analysis such as Kim et al. [2], Shiratori et al. [3], and most recently Chu and Tsai [4] agree that key dance poses occur at stops or turns in the performer's movement trajectories. Furthermore, dance movements, and human movements in general, can be segmented into primitive motions [5]. These key poses in the dance motion are said to be the motion beats. In the presence of music, the *motion beats* should be in synchrony with the rhythm of the music as defined by the *music beats* [2, 3, 6] (Fig. 1).

In the last decades, interest in techniques that post process images and videos has increased, particularly in techniques that attempt to enhance a subject captured in an image or a video: see, e.g., the works of Levyand et al. [7] and Zhou et al. [8]. In our work, dance performance enhancement is applied at the frame level without manipulating the context of the frames. The dancing performance as observed in a video is analyzed and segmented into motion intervals delimited by the motion beats. The rhythm of the dance is enhanced by manipulating the temporal domain of each motion interval using a dynamic time-warping technique similar to that used in Zhou et al. [9, 10]. The challenge is twofold: (i) extracting the motion beats from the video sequence; and (ii) defining and optimizing an objective function for synchronizing the motion beats with the music beats.

To extract the motion beats from the video sequence, we analyze at pixel level the optical flow in frames of the video; unlike the work of Chu and Ref. [4] where motion beats are tracked by object-space analysis, here the analysis is applied in image space, bypassing the difficulties incurred in tracking objects in videos. We analyze the optical flow across a range of frames to detect significant changes in

[1] Tel Aviv University, Tel Aviv 6997801, Israel. E-mail: R. Bellini, rachele.bellini@gmail.com (✉); Y. Kleiman, yanirk@gmail.com; D. Cohen-Or, dcor@tau.ac.il.

Manuscript received: 2018-02-03; accepted: 2018-03-27

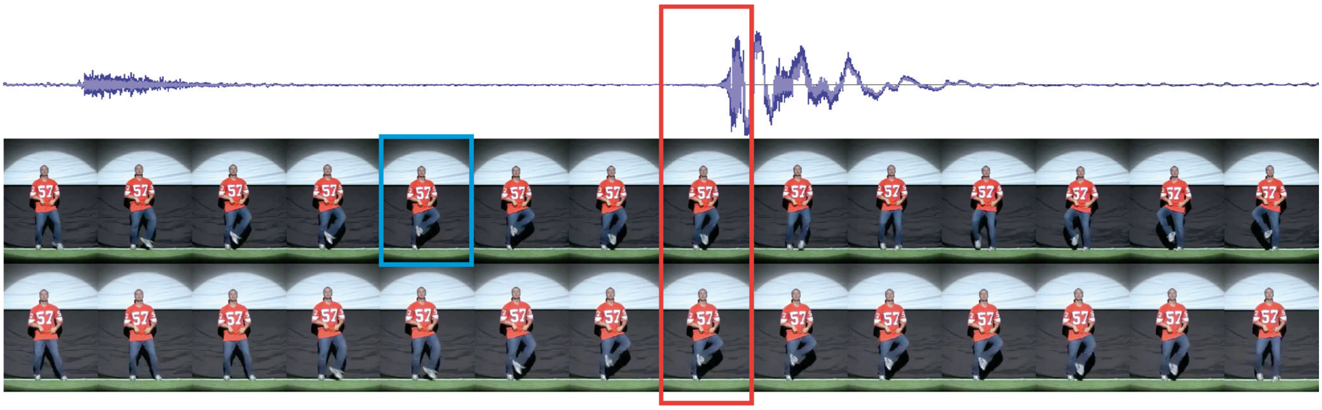


Fig. 1 A music signal (above), and a sequence of frames from both the input video (middle) and output video (below); the music beat is marked by a red frame. The output video is synchronized with the music beat, while in the input video, the motion beat, marked by a blue frame, occurs 3 frames before the music beat.

directions, which indicate locations of key poses in the video. Our method is capable of detecting fast changes in direction, as well as gradual changes, and long pauses in the dance followed by a change in direction.

Given the music and motion beat streams, we find a mapping between motion beats and music beats that minimizes the time-warping distortion of the video. We introduce an objective function that aims to maximize the number of good matches between motion beats and music beats. This optimization is constrained in the sense that matches are not enforced; only good matches are allowed, which may leave some motion beats unmatched. Using the mapping between motion beats and music beats, we adapt the video to the music using a time-warping technique, stretching or contracting each segment of the video sequence to align the motion beats with the music beats to which they are mapped.

We show two applications for this cross-media synchronization: in one, a given dance performance is enhanced to be better synchronized with its original music, and in the other, a given dance video is automatically adapted to be in synchrony with different music. A blind user study was conducted, in which users were requested to compare the input and output videos, and decide which was better synchronized with the music. The results show improvement for most videos tested, as explained in detail later. All of our input and output videos can be found in the Electronic Supplementary Material (ESM) and the online project page of this paper at `http://sites.google.com/site/dancetothebeatresults`.

2 Related work

2.1 Synchronizing video and music

Recent years have seen a vast increase in the availability of digital video and audio. This has led to a rising interest in video editing techniques for researchers and end users alike. Several recent works have been dedicated to the concept of synchronizing video files with audio files to create music videos or to enhance the entertainment value of a video. Yoon et al. [11] create a music video by editing together matching pairs of segmented video clips and music segments in a sequence. Matching is performed using features such as brightness and tempo which are extracted from the video and audio signals. Their previous work [12] matches a given video with generated MIDI music that matches the motion features of the video. The motion features are tracked with the help of user interaction. Jehan et al. [13] present an application that creates a music video for given music by using preprocessed video clips with global speed adjustment as required to match the tempo of the music.

The more recent work of Chu and Tsai [4] extracts the rhythm of audio and video segments in order to replace the background music or to create a matching music video. After the rhythm is extracted from the audio and video segments, each audio segment is matched to a video segment shifted by the constant time offset that produces the best match for the music. This method works well for the creation of a music video where a large set of video clips or audio clips is available. However, if only a single video clip and a single audio clip are available, matching is inexact. In contrast, our method processes a single video segment by locally warping its speed along the video, to produce a natural looking video that matches the given music beat by beat.

In very recent work, Suwajanakorn et al. [14] present a method to morph facial movements, matching them to an audio track. A neural network is trained to find the best match between mouth shapes and phonemes, while the head movements are warped using dynamic programming. While the results are impressive, their goal and end results are quite different to those of the work presented here, which is focused on matching a given video to a soundtrack without altering its content.

Other works involve synchronization of a pair of video sequences or motion capture data [9, 10, 15–17]. They extract a feature vector from each frame and align the dense signals, as opposed to a sparse stream of motion beats used in our work. Zhou and De la Torre [10] allow the alignment of feature sequences of different modalities, such as video and motion capture; however, their technique assumes a strong correlation between the sequences, e.g., a pair of sequences describing the same actions in video and motion capture streams. To synchronize the two streams, these methods allow the alteration of the speed of both sequences. Humans tend to be less tolerant of irregularities in music rhythm than in video appearance; therefore in our method, the music stream is kept intact to avoid irregular and unnatural sounding audio. Furthermore, we enforce additional constraints on the video to ensure noticeable changes do not occur. Wang et al. [17] find the optimal matching between two or more videos based on SIFT features, warping segments of the videos as necessary. While such features produce good results when aligning videos with similar content, they cannot be used in video and music synchronization, since they do not capture temporal events.

Lu et al. [18] present a method to temporally move objects in a video. Objects of interest are segmented in the video, and their new temporal location is found by optimizing the user's input with some visual constraints. This kind of approach, however, is focused on the overall movement of a single object, approximating it to a rigid body. In our method, instead, we aim to process complex movements, in which different parts of objects have different motions (e.g., a person dancing).

2.2 Beat extraction

Our method is based on synchronizing motion beats to music beats. Firstly, we detect motion beats from the video and music beats from the audio. There is a body of work regarding motion beat detection from motion capture data; works such as Kim et al. [2] and Shiratori et al. [3, 19] detect motion beats from motion capture data to analyze dance motion structures, in order to synthesize a novel dance or to create an archive of primitive dance moves.

Detecting motion beats from a video is a more challenging task, especially from amateur or home-made videos, which typically are contaminated with significant noise, moving background objects, camera movement, or incomplete views of the character. Chu and Tsai [4] track feature points in each frame and build trajectories of feature points over time. Motion beats are then detected as stops or changes in trajectory directions. Denman et al. [20] find motion clusters and use them to detect local minima in the amount of motion between frames.

Extraction of music beats, or *beat tracking*, is a well researched problem in audio and signal processing. The aim of beat tracking is to identify instants of the music in which a human listener would tap his foot or clap his hands. McKinney et al. [21] provide an overview and evaluation of several state of the art methods for beat tracking. These methods usually discover the tempo of the music, and use it to extract music beats that are well matched to the tempo.

The work of Ellis [22] provides a simple approach that, after discovering the tempo, uses dynamic programming to produce a series of beats which best match the accent of the music. We use the code provided by the author for our music beat tracking.

3 Motion-beat extraction

3.1 Outline

The motion beats of a dance are defined as frames in which a significant change in direction occurs. Often, there is a motion stop for a few frames in between a significant change in direction. Therefore, to detect a motion beat, we consider a non-infinitesimal time range that consists of a number of frames. A *direction-change score* is defined for every frame, and computed over multiple time ranges to detect direction changes of various speeds. A low score indicates a strong change in direction. Motion beats are then detected as local temporal minima of the frame-level score, ensuring detection of the most prominent changes in direction over time.

3.2 Super-pixel optical flow

Extracting the motion beats from a large variety of videos is a challenging task. A video can be of low quality and low resolution, such as one captured by a smartphone. We assume that the input video consists of fast motions of an object over a natural background. However, the motions of the object do not necessarily follow a fixed rhythm. Since the moving object is also not necessarily human, we do not operate in object space, but in image space.

Our technique is based on analyzing the optical flow of the video. In particular, we use the MATLAB implementation of the Horn–Schunck optical flow technique. Since the video may be noisy, we first apply a median filter and consolidate the optical flow by considering spatio-temporal super-pixels. Each super-pixel is a 5 × 5 block of pixels over five frames. We assign to each pixel the mean pixel-level optical flow over the spatio-temporal super-pixel centered around it. These super-pixel optical flow values are considerably less noisy, yet still fine enough to faithfully measure the local motion direction. Note that the motion analysis is meant to identify the motion beats at the frame level rather than at the pixel level.

The super-pixel optical flow of four frames is illustrated in Fig. 2(b), where each color depicts a different direction: blue for left and down, cyan for left and up, red for right and down, and yellow for right and up.

3.3 Frame-level motion beats

Frame-level motion beat extraction is applied at the window level. Frames are subdivided into a grid of 8 × 8 windows; we assume that in each window only one object is moving. Larger windows, or the entire frame for that matter, may contain multiple objects with opposing directions, for example a symmetrical movement of the hands. Such motion would nullify the average direction of movement. Within a small window only one object is moving, and the average direction has a meaningful value. The average directions of the windows are shown in Fig. 2(c), using the same color encoding as in Fig. 2(b).

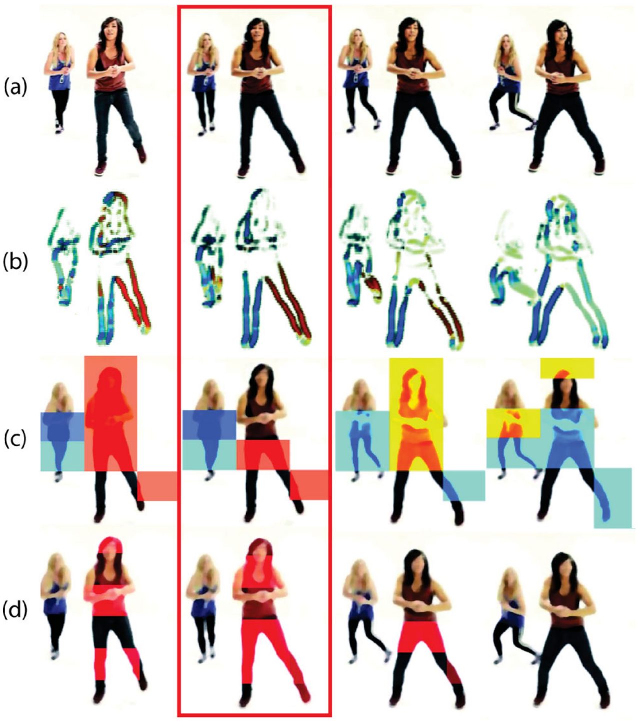


Fig. 2 Motion-beat extraction. The original video (a) is displayed along with (b) its super-pixel optical flow, (c) the average motion direction for each window, and (d) the average direction change. The detected motion beat is marked with a red outline.

We define the direction-change score as the dot product of the normalized average motion vectors of two frames; the dot product is maximal where the motion vectors have the same direction and minimal when they have exactly opposite directions. We compute the score function at the window level. To reduce temporal noise in the direction of motion, the average direction is first smoothed over time using a Gaussian filter. Since a change in direction in natural movement is often gradual, we identify changes in direction that occur at varying speeds. We compare pairs of frames at various intervals centered around each frame in the video. The score for every frame is the minimum score among all intervals centered around that frame. Figure 3 shows a temporal window around frame i. Frame $(i-6)$ is compared to frame $(i+6)$, frame $(i-5)$ to frame $(i+5)$, and so on. The dot product for each pair of frames is shown above the video frames. The direction-change score for frame i is the minimum over all pairs, which is reached in this example for the pair $\{i-4, i+4\}$.

The measurements described above provide a score for the change of direction for every spatial window; a low value suggests a strong change. The frame-level score is simply the minimal score over all windows,

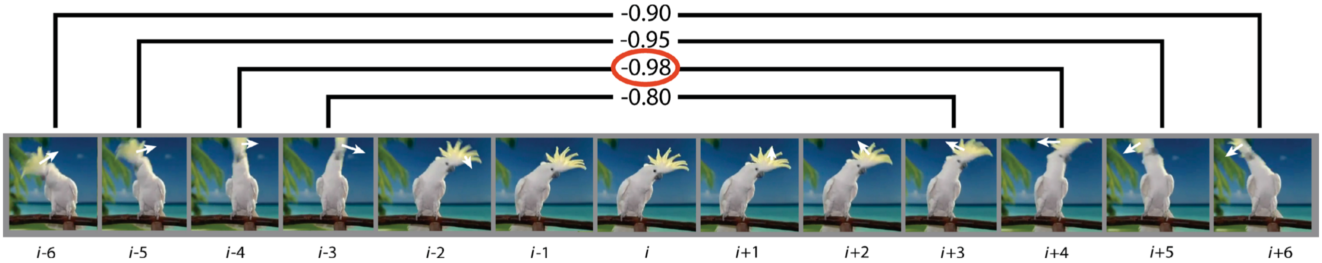


Fig. 3 Temporal window in which pairs of frames are compared. White arrows mark the approximate direction in which the parrot's head is moving. The dot product is computed for each pair of frames and the minimum is chosen as the direction-change score for that frame; the dot product is actually computed separately for each window in the 8 × 8 grid.

detecting local changes of direction over all areas of the frame. The global score for every frame in the video is displayed in Fig. 4, along with the detected local minima of the score, marked by the green vertical lines. The local minima are computed by finding the strongest negative peaks of the function that are at least five frames apart from each other. Figure 2(d) shows the color-coded score of the spatial windows in four frames. The negative values are shown in red, where a stronger color indicates a lower score, closer to -1. A motion beat is detected in the second frame from the left, which has a local temporal minimum of the frame-level change of direction score. Note that these scores are computed using a larger temporal window which is not shown in the figure.

4 Synchronization

4.1 Approach

The detected motion beats effectively divide the video into a series of *motion intervals*, each starting one frame after a motion beat and ending at the next motion beat. *Music intervals* can be defined similarly using the music beats. When a motion beat occurs on the same frame as a music beat, the dancing object appears to be in synchrony with the music. Our goal is thus to adjust the speed of each interval such that as many motion beats as possible occur on the same frame as music beats, with as little distortion as possible of each interval.

The output of the audio and video analysis is two sequences: the music beat times A_i and the motion beat times V_j. When two beats are matched, the relevant motion interval is stretched or contracted so that the motion beat occurs simultaneously with the music beat. The music signal is fixed and does not change; if the movements of the performer become slightly faster or slower, the final video can still be perceived as natural, while even the smallest modification to the music may be perceived as unpleasant. Respecting these asymmetric properties of the video and music streams, we differ from

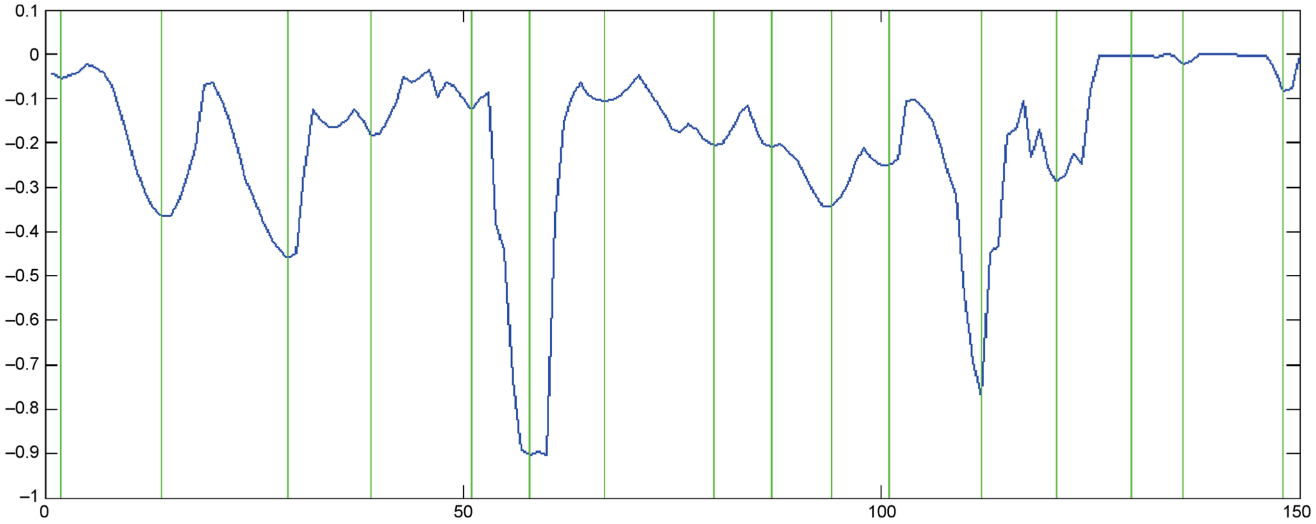


Fig. 4 Frame-level direction-change score for each frame in the video, along with detected local minima of the score (marked by green vertical lines).

previous solutions offered to related tasks by works such as Zhou et al. [9, 10], and solve an optimization problem which better fits the task at hand.

To compute the optimal mapping between motion beats and music beats, a time-warping distortion of every matched pair is measured. Naturally, not every motion beat has a matching music beat, and vice versa. However, the mapping is monotonic, i.e., every matched motion beat is at a later time than the previous matched motion beat, and likewise for music beats. Therefore, to compute the time-warping distortion incurred by a matched pair, it is necessary to accumulate all motion intervals and music intervals, respectively, between the previously matched beats and the currently matched beats. We compute a *compatibility score* for the accumulated intervals, which is high when the time-warping distortion between the accumulated intervals is low. For further details see Section 4.2.

The global optimization problem is to find the mapping which maximizes the total compatibility score over all pairs. This encourages as many matched beats as possible, as long as they do not force a large time-warping distortion for other matched pairs. Local time-warping distortions between successive pairs are also subject to user-defined constraints. The user may control and can define separate constraints for speeding up the video and for slowing it down, since different videos tolerate different amounts of speed change before losing their natural appearance, which depends on their content. Once the beats have been analyzed and matched, the video is processed by contracting or stretching motion intervals in order that the motion beats occur simultaneously with their matched music beats. The output is a time-warped version of the original video that now fits the given music.

4.2 Compatibility score

The compatibility score is defined for every possible monotonic mapping between the motion beats and music beats. It measures the total amount of distortion over all pairs of matched beats. A high compatibility score means low distortion, so maximization of the compatibility score encourages as many matching beats as possible, as explained above.

Formally, for every music beat A_i and motion beat V_j we define the lengths in frames of their corresponding intervals:

$$l_a(i) = A_i - A_{i-1}, \qquad l_v(j) = V_j - V_{j-1}$$

Throughout this section, i is the index of the current audio beat and j is the index of the current video beat. The initial intervals are

$$l_a(1) = A_1, \qquad l_v(1) = V_1$$

Every mapping is a sequence of pairs $\{i^{(m)}, j^{(m)}\}$, $1 \leqslant m \leqslant k$, where for every m the music beat $A_{i^{(m)}}$ is matched with the motion beat $V_{j^{(m)}}$.

For clarity, we define the aggregations of several intervals between two beats as

$$L_a(i_1, i_2) = \sum_{i_1 < i \leqslant i_2} l_a(i), \qquad L_v(j_1, j_2) = \sum_{j_1 < j \leqslant j_2} l_v(j)$$

Note that i_1 and j_1 can be zero, in which case aggregation starts from the first interval. The compatibility score C for an arbitrary pair of sequences starting from $\{i_1, j_1\}$ and ending at $\{i_2, j_2\}$ is then:

$$C(i_1, i_2, j_1, j_2) = \frac{\min\{L_a(i_1, i_2), L_v(j_1, j_2)\}}{\max\{L_a(i_1, i_2), L_v(j_1, j_2)\}}$$

The score $C(i_1, i_2, j_1, j_2)$ for two pairs of beats $\{i_1, j_1\}, \{i_2, j_2\}$, indicates the distortion incurred by this sequence of intervals which includes no other matching pair in between. Scores are within the range $[0, 1]$. A high value indicates low distortion; a perfect match in which the video is neither stretched or contracted has a score of one. We can now solve the global optimization problem by maximizing the total score for all pairs associated with a mapping.

4.3 Dynamic programming solution

The sequence with maximal score up to any pair of beats $\{A_i, V_j\}$ can be computed regardless of any matched pairs that are selected afterwards. We can thus solve the optimization problem by use of dynamic programming.

We define a global score function $F(i, j)$ that aggregates the values of the compatibility score for sequences up to pair $\{A_i, V_j\}$. The maximization problem for each pair is then:

$$F(i, j) = \max_{\substack{1 \leqslant k \leqslant i \\ 1 \leqslant l \leqslant j}} \{F(i-k, j-l) + C(i-k, i, j-l, j)\}$$

with initial values

$$F(0, j) = 0, \qquad F(i, 0) = 0$$

During the computation of each pair, the distortion constraints are tested, and pairs that violate the constraints are discarded. Since the maximization of F at each step is dependent on $i \times j$ values, the theoretical complexity of the computation is

$O(n_a^2\ n_v^2)$, where n_a is the number of music beats and n_v is the number of motion beats, or $O(n^4)$ if n_a and n_v are both about n.

However, many possibilities can be ignored in practice. Between two consecutive matched pairs, it is unlikely that several beats in a row will be left unmatched for both music beats and motion beats. It is possible that a corresponding pair of motion beat and music beat are both left unmatched if their distortion score violates the constraints. However, due to the regularity of the music beats and the minimum distance between motion beats, it is improbable that the difference between consecutive matched pairs will be several motion beats *and* several music beats. It is far more likely that some motion beats are left unmatched which are between two consecutive music beats or vice versa. Hence, it is sufficient to maximize $F(i, j)$ over the narrow band: $\{i', j-1\}, \{i', j-2\}, \{i-1, j'\}, \{i-2, j'\}$, where $1 \leqslant i' < i, 1 \leqslant j' < j$. This has a time complexity of $O(n_a + n_v)$ for each pair $\{i, j\}$. The total complexity of such a solution is $O(n_a n_v(n_a + n_v))$, or $O(n^3)$ if n_a and n_v are both about n.

An example of the synchronization process is shown in Fig. 5; it is shown as a sequential process for clarity. The top row shows the original music beats and motion beats for the video sequence. Music beats marked in grey squares indicate beats that are already matched; those in black squares are yet unmatched. Motion beats in the video are marked by red highlights. In the second row, the second music beat at frame 20 is matched with the second motion beat at frame 22. The video is thus pushed backwards by two frames and the position of the next motion beat is modified accordingly. Note that the score of each matched pair is based on the distance between two consecutive beats and not on the absolute position of the beat. The fourth motion beat is now at frame 54 which is closer to 50 than to 60. The score for matching the beat to frame 50 is also a little higher. However, matching the motion beat to frame 50 would require shortening the video segment by 40%. The user selected to constrain the speed ratio of the video to no more than a 25% increase in speed, so this potential match is discarded. On the other hand, the constraint for stretching the video is more loose, since slowing down the video by a certain ratio looks more natural to the viewer than speeding it up by the same ratio. Therefore, the beat is matched to the music beat at frame 60 and the video is stretched accordingly. If the user wished to also constrain stretching of the video to a lower ratio, this motion beat would not be matched to a music beat, and the next motion beat would be considered.

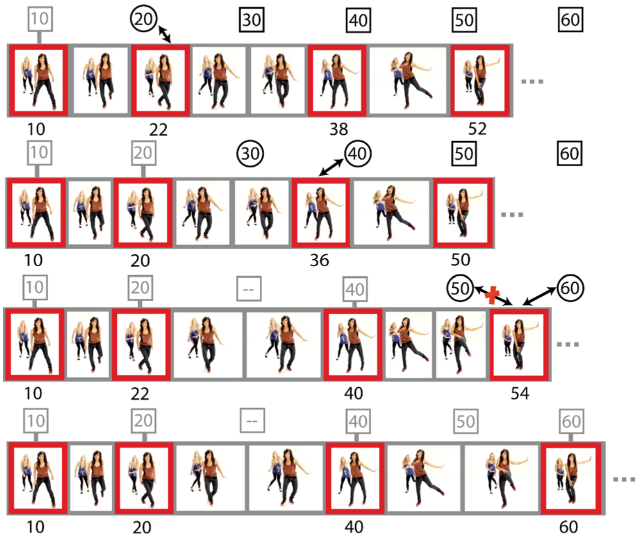


Fig. 5 Synchronization. See Section 4.3 for details.

5 Experimental results

5.1 Setting

The performance of the presented method was evaluated on a number of videos representing a variety of types of dancing, motions, rhythms, environments, and subjects. The videos also varied in quality: some videos were of commercial production levels, while others were of home video quality, captured with a hand held camera or even a smartphone, and contained significant noise. Figure 6 shows a few frames from each video to capture its essence. For further evaluation, we refer the reader to the accompanying videos in the ESM and the online project page mentioned above, where all of the examples can be found and evaluated. We experimented with two types of application as described in the following.

5.2 Performance enhancement

Here we consider a video containing a dance or rhythmical movements, which does not match the background music perfectly. The original video is then edited by our method, and the dancing enhanced to better fit the beats of the original music: the motion beats are adjusted to match the music beats.

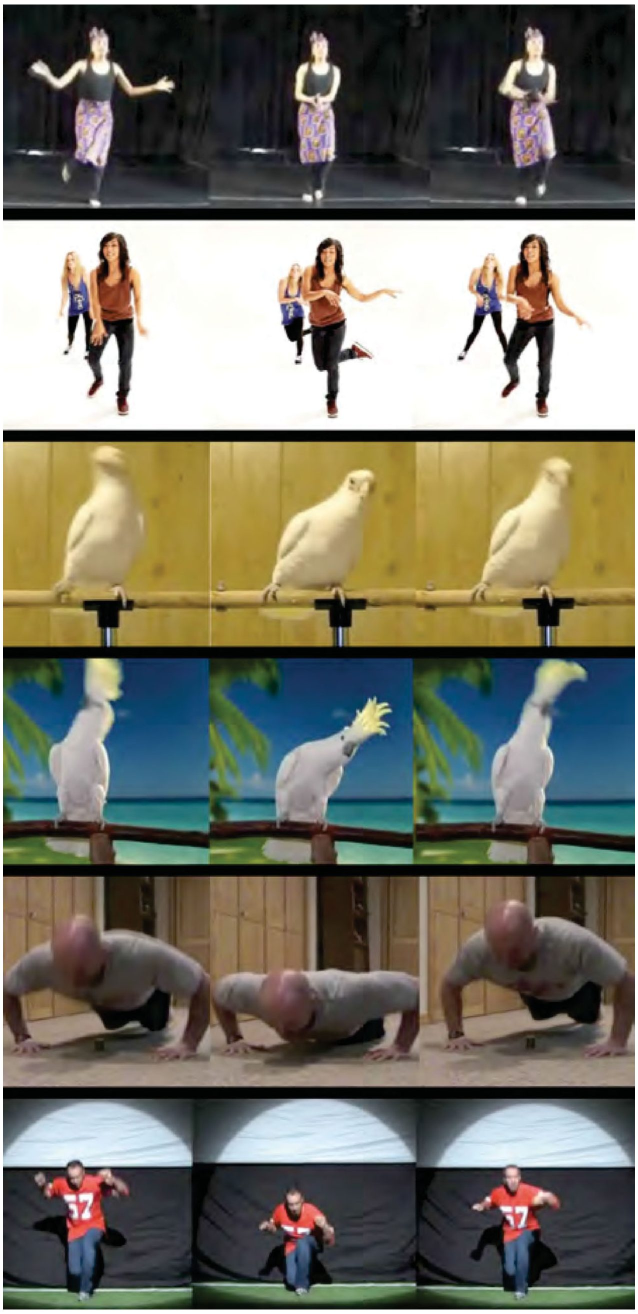


Fig. 6 Typical frames from each input video; the center frame shown for each video occurs at a motion beat. Note that we can handle videos with a moving camera (such as the first video) and significant motion blur (such as the parrot videos).

Firstly, we present two rather simple examples, in which the enhancement can be clearly appreciated. The performing subject in the first example is a well-known parrot named Snowball, a sulphur-crested cockatoo whose renowned dancing skills have been studied and described in several academic papers [23–25]. Although the parrot has an extraordinary ability to move according to the music beat, its performance is still imprecise. Snowball is famous enough to have been cast for a Taco Bell commercial in 2009, where he dances along with the song "Escape (the Piña Colada song)" by Rupert Holmes. Some of the movements of the parrot in the video are irregular, so some motion beats are not synchronized with the music beats. As can be observed, our method modifies the video so that the movements become more rhythmical and better synchronized with the given song.

The subject of the second video is another type of parrot, a bare-eyed cockatoo, which dances to the song "Shake a Tail Feather" by Ray Charles. The parrot does not have a constant rhythm, and misses the music beats quite often. Our method greatly enhances its performance.

The third example is of a human dancer performing in front of a live audience. The segment is a part of a performance by Judson Laipply called "The Evolution of the Touchdown Dance", which includes memorable NFL touchdown dances. The dancer mostly keeps the beat quite well, but in the second part of the video segment the contribution of our method is clear. It should be noted that our method does not interfere with the original video where it is well synchronized. Therefore, in this particular example, the first half of the output video looks almost the same as the input, while in the second half there is a noticeable difference.

5.3 Dance transfer

The second set of experiments transfers a dance to new background music. The input dance is modified to match the new rhythm and audio beats of the new music. Again, we begin with a few simple and clear examples. The first is a video of a man doing push-ups. After a while he gets tired and does not keep a constant rhythm in the original video. This is particularly evident around the 15th second. We synchronized it to the song "Where is the Love?" by The Black Eyed Peas. The song has a slow rhythm, so the beginning is slowed down: the input video shows ten push-ups in the first ten seconds, while the output shows only eight. However, the same rhythm is kept until the end of the video, and the man does not seem to get tired as in the original video. When synchronizing the same input video to the song "Billie Jean" by Michael Jackson, the results differ significantly. The rhythm of the song is somewhat faster, and thus the push-ups happen at a faster rate of ten push-ups in the first ten seconds. The rhythm of the push-ups is again more regular than in the input video. Note that since the rate of the push-ups is slower as the video progresses, a music beat is skipped once in a while, and a motion beat is matched with the next music beat, while still maintaining the constant rhythm.

Another clear example is given by the same Taco Bell commercial used for the first type of experiment. This time the background song is changed to the song “Trashin' the Camp” by Phil Collins. In the original version the movements of the parrot do not fit with the rhythm of the music, since they are shifted in time and their rhythm is rather faster than the music rhythm. In the output video the movements are slowed down a little and time-shifted to match the new music.

We also show a few experiments with videos of professional human dancers. The first experiment with a human dancer uses a video of Caren Calder, a dance instructor, performing a traditional West African dance. We synchronize her dance to “Oh! Darling” by The Beatles, which is a famous song from a very different culture. In the original video the dancer's hand clapping and music beats are not matched, while in the new version her movements and hand clapping occur to the beat of the music.

Another dance experiment is conducted again with footage from the performance of Judson Laipply, “The Evolution of the Touchdown Dance”, using two different segments of this performance. One has been synchronized to “Billie Jean” by Michael Jackson, and the other to “Pantala Naga Pampa” by The Dave Matthews Band. With the first song the changes are very clear, since the original video does not match the music. With “Pantala Naga Pampa”, as in the experiment using the same video segment with the original music, the dancer's movements match the music quite well, and the changes in some parts of the dance are not easily noticeable. However there are a few changes: at the very beginning, in the original video, the first jump occurs within the silence before the beginning of the song, while in the new version the first jump occurs on the first beat of the song. In the second part of the video the dancer jumps a few times with irregular rhythm. This part is significantly changed to better fit the new music.

The last video we present here is a jazz-funk dance online video of the Dr Pepper Cherry YouTube Dance Studio. The original background music is replaced and the dance is synchronized with the song “Hayling” by FC Kahuna. The original dance is quite fast, while the music we chose has very few beats per minute. In this case, since the dancers keep a rather constant rhythm, the original performance fits almost perfectly to the new music. Thus, the changes are minute and hard to perceive without careful observation of the input and output videos side by side. The user study shows most people believe the output video is better synchronized with the music, even though the differences between them are small.

We refer the reader again to the ESM or the project page to view the example videos described above.

5.4 User study

A blind user study was conducted for each of the videos described above, to which 26 users replied. Users were presented with pairs of videos, and were requested to rate which of the videos better matched the music, without knowing which video was the enhanced version and which was the original. The results of the user study are displayed in Fig. 7. The first three videos, labeled *Parrot 1*, *Parrot 2*, and *Touchdown*, were examples of performance enhancement of two different parrots and a segment from “The Evolution of the Touchdown Dance”. In these videos the background music remained the same as in the original video. The remaining videos were examples of music replacement.

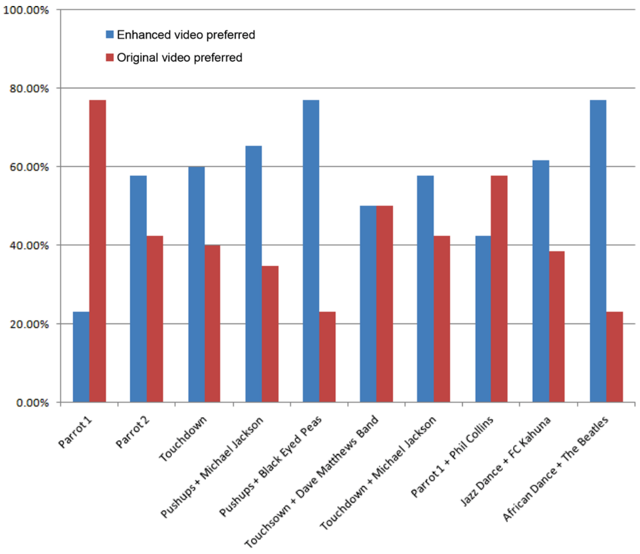


Fig. 7 User preferences when comparing the original version to the enhanced version of each video.

As can be seen from the figure, the enhanced videos were usually preferred to the corresponding input videos. Our most successful examples were ones in which the dance performance deviated significantly from the rhythm of the song. The African dance video has a slow rhythm, and so does the song we matched it to. Therefore users can easily identify where a music beat is missed in the original performance, and it is easier to evaluate the differences between the videos. In the push-up videos, which also received positive reviews, the person working out becomes tired and does not keep a constant rhythm, which again enables the user to clearly see where a music beat is missed.

Conversely, for video segments which maintain a fast fixed rhythm, it is harder for the human mind to distinguish between different motion beats and separate them from the rhythm of the attached music. Even a professional video editor may require several viewings to determine whether a certain motion beat occurs simultaneously with a music beat. However, it can often be felt subconsciously whether a certain video is in sync with the music or not. An interesting example is the jazz dance performance; the rhythm of the dance is very fast, and the replaced music matches the original video quite well. Only minor corrections are applied by our synchronization algorithm, and the output video looks almost identical to the the original video: the differences between the videos are only noticeable when the videos are viewed simultaneously side by side. Yet in our user study, more than 60% of the users preferred the enhanced video to the original.

A noticeable failure shows that sometimes keeping the beat is not enough to present a good dance performance. In the video of the parrot Snowball, the head motions do not follow the beat particularly well. However, the parrot changes his dancing style noticeably as the chorus of the song begins. In our enhanced version, the individual motions of the parrot's head are better synchronized with the music, but the change of dancing style occurs a few music beats later than the beginning of the chorus. The users did not react well to that change and preferred the original performance of the parrot. A possible improvement to our method in such cases is to synchronize motion beats to music beats only within segmented non-overlapping sections of video.

5.5 Implementation

Our algorithm was implemented in MATLAB. We ran all experiments on an Apple MacBook Pro with a 2.4 GHz Intel Core 2 Duo processor and 4 GB of RAM, and the Mac OS X 10.6.8 operating system. The experiments were mostly run on videos of about 500 frames, or 20 seconds, with 640×360 resolution. The bottleneck in our implementation is the video editing itself, i.e., reading and writing video frames, and video compression. This could be vastly improved by using other video compression methods (e.g., an image sequence) and faster editing methods. We estimate that the algorithm itself, without the video editing, i.e., detecting the motion beats and music beats and finding an optimal mapping between them, requires about two minutes for each video above.

6 Conclusions and future work

We have presented a method to enhance a video of a dancing performance. The key idea is to synchronize the motion beats to the music beats so that the dance rhythm is better correlated with the given music. Detected motion beats are used to delimit motion intervals. However, while music beats are well understood, the notion of motion beats is not yet fully established; there are other possible definitions for motion beats, e.g., using the centers, rather than the ends, of motion segments. How to segment motion in video remains an open interesting problem. Clearly, segmenting motion of articulated bodies can be performed better in object-space. However, this requires identifying and tracking the skeleton of the dancing body, which is known to be a difficult task. In our work, we analyze the video frames directly, detecting motion beats in image-space. The main advantage of our approach is that it is not limited to human bodies, or tailored to a particular subject known a priori. However, in cases of extreme camera movements, some motion beats may not be detected. As mentioned in Section 3.2, we assume that the foreground objects in the input videos move more quickly than the background. In future, we aim to consider integrating object-space analysis, to see whether it can significantly improve the segmentation of the dancing subject.

Another interesting direction for future work is intra-frame enhancement of videos containing more than a single dancing performer. Here we can use the extracted motion beats, and synchronize two or more motions in the video with each other as well as with the background music. In this case, the correct synchronization will be more evident than in a cross-media synchronization.

The motion beats we detect can also be helpful outside the task of synchronization; one such possible application is global time remapping for a video. When stretching or contracting a video, deformation can be concentrated around the motion beats. Video segments where continuous motion occurs would retain their natural speed, while motion stops, or segments where there is a change in direction, could be prolonged or shortened. Since the speed of motion is low in such segments, the change of speed would be less noticeable and more natural looking.

Electronic Supplementary Material Supplementary material is available in the online version of this article at https://doi.org/10.1007/s41095-018-0115-y.

References

[1] Repp, B. H. Musical synchronization. In: *Music, Motor Control and the Brain.* Altenmuller, E.; Wiesendanger, M.; Keselring, J. Eds. Oxford University Press, 55–76, 2006.

[2] Kim, T.-h.; Park, S. I.; Shin, S. Y. Rhythmic-motion synthesis based on motion-beat analysis. *ACM Transactions on Graphics* Vol. 22, No. 3, 392–401, 2003.

[3] Shiratori, T.; Nakazawa, A.; Ikeuchi, K. Dancing-to-music character animation. *Computer Graphics Forum* Vol. 25, No. 3, 449–458, 2006.

[4] Chu, W.-T.; Tsai, S.-Y. Rhythm of motion extraction and rhythm-based cross-media alignment for dance videos. *IEEE Transactions on Multimedia* Vol. 14, No. 1, 129–141, 2012.

[5] Flash, T.; Hogan, N. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience* Vol. 5, No. 7, 1688–1703, 1985.

[6] Jones, M. R.; Boltz, M. Dynamic attending and responses to time. *Psychological Review* Vol. 96, No. 3, 459–491, 1989.

[7] Leyvand, T.; Cohen-Or, D.; Dror, G.; Lischinski, D. Digital face beautification. In: Proceedings of the ACM SIGGRAPH 2006 Sketches, Article No. 169, 2006.

[8] Zhou, S.; Fu, H.; Liu, L.; Cohen-Or, D.; Han, X. Parametric reshaping of human bodies in images. *ACM Transactions on Graphics* Vol. 29, No. 4, Article No. 126, 2010.

[9] Zhou, F.; Torre, F. Canonical time warping for alignment of human behavior. In: Proceedings of the Advances in Neural Information Processing Systems, 2286–2294, 2009.

[10] Zhou, F.; De la Torre, F. Generalized time warping for multi-modal alignment of human motion. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1282–1289, 2012.

[11] Yoon, J.-C.; Lee, I.-K.; Byun, S. Automated music video generation using multi-level feature-based segmentation. In: *Handbook of Multimedia for Digital Entertainment and Arts.* Furht, B. Ed. Springer, 385–401, 2009.

[12] Yoon, J.-C.; Lee, I.-K.; Lee, H.-C. Feature-based synchronization of video and background music. In: *Advances in Machine Vision, Image Processing, and Pattern Analysis. Lecture Notes in Computer Science, Vol. 4153.* Zheng, N.; Jiang, X.; Lan, X. Eds. Springer Berlin Heidelberg, 205–214, 2006.

[13] Jehan, T.; Lew, M.; Vaucelle, C. Cati dance: Self-edited, self-synchronized music video. In: Proceedings of the ACM SIGGRAPH 2003 Sketches & Applications, 1–1, 2003.

[14] Suwajanakorn, S.; Seitz, S. M.; Kemelmacher-Shlizerman, I. Synthesizing Obama: Learning lip sync from audio. *ACM Transactions on Graphics* Vol. 36, No. 4, Article No. 95, 2017.

[15] Caspi, Y.; Irani, M. Spatio-temporal alignment of sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 24, No. 11, 1409–1424, 2002.

[16] Slaney, M.; Covell, M. FaceSync: A linear operator for measuring synchronization of video facial images and audio tracks. In: Proceedings of the 13th International Conference on Neural Information Processing Systems, 784–790, 2000.

[17] Wang, O.; Schroers, C.; Zimmer, H.; Gross, M.; Sorkine-Hornung, A. VideoSnapping: Interactive synchronization of multiple videos. *ACM Transactions on Graphics* Vol. 33, No. 4, Article No. 77, 2014.

[18] Lu, S.-P.; Zhang, S.-H.; Wei, J.; Hu, S.-M.; Martin, R. R. Timeline editing of objects in video. *IEEE Transactions on Visualization and Computer Graphics* Vol. 19, No. 7, 1218–1227, 2013.

[19] Shiratori, T.; Nakazawa, A.; Ikeuchi, K. Detecting dance motion structure through music analysis. In: Proceedings of the 6th IEEE International Conference on Automatic Face and Gesture Recognition, 857–862, 2004.

[20] Denman, H.; Doyle, E.; Kokaram, A.; Lennon, D.; Dahyot, R.; Fuller, R. Exploiting temporal discontinuities for event detection and manipulation in video streams. In: Proceedings of the 7th ACM SIGMM

International Workshop on Multimedia Information Retrieval, 183–192, 2005.

[21] McKinney, M. F.; Moelants, D.; Davies, M. E. P.; Klapuri, A. Evaluation of audio beat tracking and music tempo extraction algorithms. *Journal of New Music Research* Vol. 36, No. 1, 1–16, 2007.

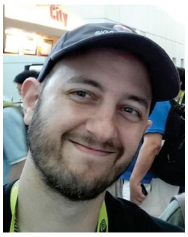
[22] Ellis, D. P. W. Beat tracking by dynamic programming. *Journal of New Music Research* Vol. 36, No. 1, 51–60, 2007.

[23] Patel, A. D.; Iversen, J. R.; Bregman, M. R.; Schulz, I. Experimental evidence for synchronization to a musical beat in a nonhuman animal. *Current Biology* Vol. 19, No. 10, 827–830, 2009.

[24] Patel, A. D.; Iversen, J. R.; Bregman, M. R.; Schulz, I. Studying synchronization to a musical beat in nonhuman animals. *Annals of the New York Academy of Sciences* Vol. 1169, No. 1, 459–469, 2009.

[25] Patel, A. D.; Iversen, J. R.; Bregman, M. R.; Schulz, I.; Schulz, C. Investigating the human-specificity of synchronization to music. In: Proceedings of the 10th International Conference on Music Perception and Cognition, 100–104, 2008.

Rachele Bellini received her B.Sc. degree cum laude in digital communication in 2012 and her M.Sc. degree cum laude in computer science in 2015, both from the University of Milan. Since 2014 she is collaborating with Cohen-Or's group at Tel Aviv University on texturing, image processing, and video analysis. She is currently working as a VFX Developer at Pixomondo LLC (Los Angeles).



Yanir Kleiman obtained his Ph.D. degree from Tel Aviv University in 2016, under the supervision of Prof. Daniel Cohen-Or. He was a post-doctor at the École Polytechnique in France in 2016 and 2017. He is currently a graphics software developer at Double Negative Visual Effects. His research focuses on shape analysis, including shape similarity, correspondence and segmentation, and included work in other domains such as image synthesis, crowd sourcing, and deep learning. Prior to his Ph.D., Yanir had more than 10 years of professional experience as a software developer and visual effects artist.

Daniel Cohen-Or is a professor at the School of Computer Science, Tel Aviv University. He received his B.Sc. (cum laude) degree in mathematics and computer science and his M.Sc. (cum laude) degree in computer science, both from Ben-Gurion University, in 1985 and 1986, respectively. He received his Ph.D. degree from the Department of Computer Science at the State University of New York at Stony Brook in 1991. He received the 2005 Eurographics Outstanding Technical Contributions Award. In 2015, he was named a Thomson Reuters Highly Cited Researcher. Currently, his main interests are in image synthesis, analysis and reconstruction, motion and transformations, and shapes and surfaces.