Fully Automatic and Frame-accurate Video Synchronization using Bitrate Sequences

Florian Schweiger, Georg Schroth, Michael Eichhorn, Anas Al-Nuaimi, Burak Cizmeci, Michael Fahrmair and Eckehard Steinbach

***Abstract*—Video synchronization is an essential processing step in many multimedia applications, and various methods have been proposed in the literature each of which addresses the problem from a different point of vantage. In this article, we present an information theoretic approach to video synchronization, based on the state-of-the-art in hybrid video coding. Time series derived from the videos' instantaneous bitrate demand are correlated in a robust manner employing the recently published ConCor algorithm. We enhance ConCor with integrated normalization capabilities in order to improve its shape-oriented matching performance. Furthermore, we present a mathematical framework to derive the most suitable ConCor parameters given a specific class of input videos. In an extensive experimental analysis, we give an insight into the representation of synchronization-relevant scene changes with bitrate data, and examine the influence of encoding parameters on the synchronization performance. Experiments on diverse video input substantiate the reliable performance of our easy to implement, yet effective video synchronization algorithm which distinguishes itself in that it operates largely without manual intervention.**

I. Introduction

Multimedia applications involving multiple videos of the same scene typically require exact temporal synchronization in order to extract useful information from the given data. For instance, the inference of depth information from multi-view videos can in general only be performed if the corresponding frames have been acquired at the same time instant. Otherwise, the projections of dynamic objects are inconsistent. The same holds for applications which aim at stitching the input videos together to form a panoramic view, or at editing videos in such a way as to provide seamless transitions between different perspectives on the portrayed action. Fields of application are very diverse and include basically every domain where multiple cameras are deployed. Be it any kind of camera network, used for instance in a surveillance application, in television or film production, or novel community based video sharing applications; whenever two or more videos of the same scene are available, there is an interest in aligning the image sequences in time. The actually required precision of this alignment may vary depending on the particular application. Typically a synchronization with integer frame accuracy or below is desired. Another important requirement in many applications is that the synchronization process should be fully automatic, without the need for user intervention.

While the above demands can be satisfied with hardware-based solutions, such approaches are not applicable in many of the targeted scenarios. In fact, cameras related by a central clock are only used in high-end applications, such as professional multi-view sportscasts or automotive crash tests. An alternative are deliberately placed, external synchronization cues, *e.g.*, by use of a clapper board in film productions. None of these approaches are practical unless the camera setup is permanent to some extent, or the acquisition is planned well in advance. For less costly productions, or in scenarios where a camera network is formed in an ad-hoc manner, possibly without control over the used cameras, only software-based synchronization approaches are viable. In the case of user generated content, an event might have been captured independently by strangers, their recordings only unified afterwards through some common video sharing service [1].

The less influence one has on the acquisition process, the more robustness a video synchronization algorithm needs to offer. Assumptions about perfectly stationary cameras, identical frame rates, or similar viewing directions are more often than not invalid in practice. Ideally, a video synchronization algorithm can deal with unknown input sequences and, as long as they show the same event, reliably determine their temporal alignment.

In this article we describe a fully automatic and frame-accurate video synchronization approach that is based on bitrate characteristics and hence largely independent of acquisition properties. The approach builds on our findings from previous publications. Starting from the initial discovery in [2] that bitrate profiles are a powerful source of information for video synchronization, we proposed a robust video synchronization algorithm based on Consensus-based Cross-correlation (ConCor) in [3]. In this article, we extend our preliminary work from [2] and [3] in several directions. We specifically describe how to extend ConCor with a normalization scheme so as to improve the alignment as a whole. Furthermore, we present a sophisticated error model that directly leads to the optimal choice of the ConCor segmentation parameters. Moreover, we analyze the re-encoding process in general and derive practical guidelines for synchronization-related bitrate extraction based on H.264/AVC. Compared to [2] and [3], the extensions described in this paper lead to a substantial performance improvement. For the tested datasets, in comparison with [3], the percentage of videos perfectly synchronized rises from 50% to 67%. The percentage of videos synchronized with accuracy within one frame improves

Part of this work has appeared in preliminary form at IEEE ICIP 2010 and ACM Multimedia 2011.

The authors are with the Institute for Media Technology at Technische Universität München, Munich, Germany; email: {florian.schweiger, schroth, michael.eichhorn, anas.alnuaimi, burak.cizmeci, eckehard.steinbach}@tum.de

M. Fahrmair is with DOCOMO Euro-Labs, Munich, Germany; email: fahrmair@docomolab-euro.com

from 75% to 100%.

This paper is organized as follows. After an overview of related work in Section II, we will proceed to the core idea of describing the temporal characteristics of a video sequence with the information content of its frames, measured by the instantaneous bitrate demand. Section IV is devoted to consensus-based cross-correlation (ConCor), a useful tool to deal with failures in obtaining a reliable information estimate. We incorporate normalization into ConCor to increase the overall matching performance, and furthermore derive the optimal choice for its parameters. In Section VI, we present an extensive experimental analysis of the bitrate generation process based on H.264/AVC, and discuss several details relevant for the application of our approach in practice. Finally, the synchronization performance is validated with experiments on a variety of video datasets.

II. State-of-the-art

In the late 1990s, when more and more applications involving multiple videos of one scene began to emerge, the first software-based video synchronization algorithms were proposed. Among the first authors to take on this subject are Reid & Zisserman [4] and Stein [5] who presented similar *feature-based* approaches assuming coplanar object motion. Two sequences are brought to alignment by estimating a homography between the views, the estimation error being a measure of asynchrony. Most subsequent feature-based methods have seized this fundamental principle of establishing geometric consensus between dynamic features. To deal with more general object motion, other approaches quantify misalignment by means of epipolar geometry, estimating fundamental matrices [6]–[10] or trifocal tensors [11], [12]. Rao et al. [13] and Tresadern & Reid [14] avoid the explicit computation of epipolar geometry and evaluate rank constraints instead. Some authors apply voting schemes to find the most consistent temporal alignment among feasible candidates. Pooley et al. employ the Hough Transform to establish an affine relationship between timelines [6], Pádua & Carceroni et al. use RANSAC instead [10]. Tuytelaars and Van Gool have detached their approach from epipolar constraints and evaluate the distance between back-projected rays of sight in affine space [15]. Under similar assumptions, Wolf et al. [16] solve the synchronization problem using rank constraints on the flow of tracked features, adopting principles originally presented by Irani [17]. In [18], Yan and Pollefeys extract spatio-temporal interest points from the videos and cross-correlate their occurrence over time. Raguse and Heipke presented an approach aiming at accurate alignment of footage acquired with multiple high-speed cameras [19]. They regard the cameras' temporal deviations as additional intrinsic parameters, and so incorporate them into regular bundle adjustment. Wedge et al. and Brito et al. have proposed dedicated algorithms, respectively, to synchronize recordings of objects in free fall [8], and of mobile sensors actively tracking their own position [9] (adopting the principles from [10]). Relying on external aids, these two approaches can be considered on the verge to hardware-based methods.

A second major branch of synchronization approaches is formed by *intensity-based* methods. Instead of matching image features and their trajectories, constraints on the alignment are derived from the body of pixels in all video frames. In 2000, Caspi & Irani presented their work on sequence-to-sequence alignment [20] where the temporal alignment between frames and their spatial transformation is solved for simultaneously. In an iterative approach operating on scale pyramids generated from the input videos, the actual deviation in gray levels is minimized. In later publications, a different similarity measure replacing mean squared error was introduced [21], as well as a feature-based variant of the initial algorithm [7]. Another early work by the same authors deals with the synchronization of rigidly linked, moving cameras [22], exploiting similar changes over time in both views. Dai et al. have seized the principle behind [20], but solve for the spatio-temporal alignment through 3-D phase correlation [23]. Along completely different lines, Ushikazi et al. derive a frame-wise measure of appearance change that can be matched using cross-correlation [24]. Recently, Shresta et al. published a multi-modal approach exploiting audio fingerprints and the occurrence of camera flashes which they align across videos using dynamic programming.

All these methods have their specific strengths and limitations in terms of requirements to be imposed on the cameras, their setup, and the portrayed scene itself. In particular, there are differences in the number of supported cameras, their relative orientation, the nature of allowed motion, as well as image resolutions, frame rates, etc.. Scene objects sometimes must be sufficiently textured, so as to detect, track and match reliable features, their movements restricted both in nature and intensity. Generally speaking, a main issue of feature-based approaches is their restriction of relative camera viewing angles due to limited matchability [25]. If features are derived from silhouettes, view points are typically confined to a plane, depending on the assumptions on object shapes and motion (*e.g.*, horizontal baselines in case of upright posture). Intensity-based methods on the other hand tend to be incompatible with independently moving cameras.

III. Bitrate-based Video Synchronization

In [2], we have presented a fundamentally different approach to video synchronization. Instead of imitating the human eye in detecting synchronous events in two videos, our approach depends on a more abstract concept: the information content of individual video frames. Based on the fundamental understanding that synchrony is inextricably linked with motion in the scene, it is our goal to reliably quantify motion throughout a video. Obviously, static scenes do not carry any information from one frame to the next. From an information theoretic point of view, a frame that does not differ from its neighbors exhibits vanishing conditional entropy, its additional information content is zero. Only if there is an unpredictable deviation from previous observations, a frame brings about an information increase; which is exactly the case if objects in the scene are in motion.

There have been several advances towards quantifying scene changes specifically for the purpose of video synchronization,

e.g., in [24]. The biggest challenge for all these approaches is to make the proposed measures as robust as possible to detrimental effects. In order to avoid restrictions to be imposed on the videos, the measures need to be designed to deal with all kinds of external influences: an incessantly complex task. The most important issue is to reliably distinguish between camera motion and scene motion. While the latter carries precious information closely related to synchrony, camera motion is entirely independent, thus irrelevant for synchronization (unless the cameras are rigidly linked to each other, *e.g.*, on a stereo rig).

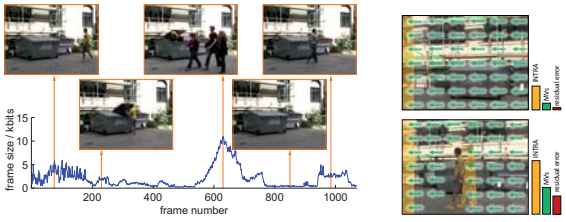
A field where this same problem – viewed from a different perspective – has already been solved to a great extent is video compression. Here, the goal is to represent video data with the least possible rate, hence to reduce it to its very essential information content. State-of-the-art video compression algorithms efficiently compensate for predictable changes, reducing the bitrate demand of corresponding macroblocks to a minimum. Fig. 1a shows the bitrate sequence obtained from a real video. Clearly, increased bitrate demand coincides with the occurrence of motion in the scene. Fig. 1b schematically illustrates the handling of scene changes by a hybrid video codec. For homogeneous motion patterns, which are characteristic for camera pans, prediction from previous frames is highly efficient. The merely translational displacement of macroblocks is encoded in motion vectors, achieving vanishing residual errors. Such a smooth motion vector field can further be represented at very low rate owing to differential coding. The major bitrate contribution in this example stems from image parts that are uncovered on the left image border due to the camera motion. Ordinarily, these macroblocks need to be encoded independently. In the case of scene motion, the bitrate composition is different, as depicted at the bottom of Fig. 1b. Not only do moving objects uncover additional background areas which lead to more independently encoded blocks. The motion vector field is also less regular, and thus more difficult to compress. Since object motion is in general more complex, prediction efficiency is also lower, leading to higher residual errors. To summarize, camera motion can be represented very efficiently while scene changes require higher bitrates. Our basic synchronization approach is to derive sequences $a(t)$ and $b(t)$ counting the number of bits necessary to encode frame t of each video, and to apply zero-mean normalized cross-correlation (ZNCC) to determine the temporal offset Δt^*:

$$\tilde{c}(\Delta t) = \frac{1}{L_b} \sum_t \frac{a(t+\Delta t) - \overline{a}(\Delta t)}{\sigma_a(\Delta t)} \cdot \frac{b(t) - \overline{b}}{\sigma_b}, \qquad (1)$$

$$\Delta t^* = \arg\max_{\Delta t} \tilde{c}(\Delta t),$$

where L_x, $\overline{x}$ and σ_x stand for the length, mean and standard deviation of a sequence $x(t)$. For more details, especially on required and optional preprocessing steps, the reader is referred to the original publication [2].

Even though camera motion cannot be fully eliminated in practice, its contribution is limited and, most notably, not correlated between views. We have shown in [2] that cross-correlating bitrate sequences is a reliable way to determine



(a) (b)

Figure 1. (a) Example bitrate profile illustrating the dependence of bitrate demand on scene motion. (b) Simplified qualitative view on bitrate contributions (INTRA encoded macroblocks, differentially encoded motion vectors, residual prediction error) for the cases of sheer camera motion (top) and additional scene changes (bottom).

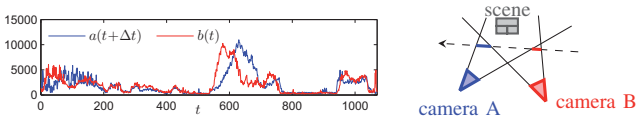


Figure 2. Two aligned bitrate sequences from the scene in Fig. 1a (left) and a schematic top view showing the corresponding camera setup (right). The incongruent peaks at $t = 600$ are caused by passers-by which, traversing the scene along the depicted trajectory, appear first in camera B's field of view.

the temporal offset between two videos of the same scene. We have further demonstrated that the bitrate-based approach imposes only minimal restrictions on the videos to be synchronized. Owing to the sophisticated motion compensation qualities of H.264/AVC [26], it can cope with moderate camera motion, and it is independent of the cameras' viewing directions since the amount of motion in the scene is quantified rather than its precise appearance. The only prerequisites are the presence of motion in the observed scene (obviously), and that the depicted actions of interest remain in the focus of both cameras throughout the recording. Fig. 2 illustrates the difficulties encountered when this last requirement is not met. The group of passers-by visible around frame 600 in Fig. 1 trigger a massive bitrate peak which occurs slightly delayed in the two videos due to the given camera setup. Because of its dominance, this peak can lead to misalignment and should thus be excluded from the correlation measure. An automatic means of doing so is ConCor [3] which will be briefly reviewed in the next section. In addition, we describe in Sections IV-B and IV-C extensions to [3] which improve the synchronization robustness significantly:

- Normalized ConCor is proposed in IV-B to mitigate the negative influence of non-stationary effects (*e.g.*, drastic changes in scene activity in the course of a recording) on the alignment performance.
- Optimal segmentation parameters are derived for ConCor in IV-C such that, from the available bitrate data, a maximum of useful information is considered during synchronization.

IV. Consensus-based Cross-correlation

As demonstrated in [2], cross-correlating bitrate sequences is a simple, yet reliable approach to determine the temporal alignment of a pair of videos. In [2], the template that was to be matched within the longer bitrate sequence had been

manually selected from the shorter one. In order to be truly independent of user intervention, an automatic mechanism is necessary to select the most suitable parts from the sequences. This mechanism needs to discard parts of the second bitrate sequence $b(t)$ which do not overlap with the first one $a(t)$. Furthermore, it should identify those signal parts where non-stationary disturbances occur, either in $b(t)$ itself, or in corresponding parts of $a(t)$. Such disturbances can be of very different causes, including temporary occlusions present in one of the views (*e.g.*, due to the passer-by effect discussed towards the end of Section III), or sudden movements of one of the cameras that cannot be fully compensated by the video codec. Another effect that renders specific signal parts useless for synchronization is, *e.g.*, the temporary lack of motion altogether.

With *consensus-based cross-correlation (ConCor)*, we have devised an algorithm that specifically addresses these requirements [3]. ConCor can detect unapt signal parts, and exclude them from the computation of the cross-correlation measure. In the following, we will briefly recall the concepts behind ConCor, develop an extension based on the ZNCC, and finally derive optimal values for ConCor's most significant parameters.

A. The Basic ConCor Algorithm

The basic idea behind consensus-based cross-correlation is to split one of the signals into shorter segments, and to cross-correlate each of them independently with the second signal:

$$c_i(\Delta t) = \sum_t a(t+\Delta t)\, b_i(t),$$
$$\text{where } b_i(t) = \begin{cases} b(t) & : (i-1)M \le t < iM \\ 0 & : \text{else} \end{cases}$$

The $c_i(\Delta t)$ are referred to as *partial cross-correlation functions* (PCCFs). Their sum is obviously equivalent to the cross-correlation function of the original sequences $a(t)$ and $b(t)$:

$$\sum_i c_i(\Delta t) = \sum_i \sum_t a(t+\Delta t)\; b_i(t)$$
$$= \sum_t a(t+\Delta t) \sum_i b_i(t) = \sum_t a(t+\Delta t)\; b(t) = c(\Delta t)$$

By omitting in the above summation those PCCFs that have been computed from erroneous or otherwise unsuitable signal parts $b_i(t)$, their influence on the resulting cross-correlation function can be specifically excluded. In [3], we describe how to apply RANSAC to make this selection and present a video synchronization example.

B. Normalized ConCor

In [3], as well as in the previous section, ConCor was proposed as a robust extension to regular cross-correlation. Neither the PCCFs $c_i(\Delta t)$ nor their combinations had been normalized with respect to the partial signals' means and standard deviations. And yet adaptive normalization is indispensable in order to gain independence of absolute signal magnitudes, thus to emphasize true shape similarities between signals. While zero-mean normalized cross-correlation (ZNCC) was the basis for our original bitrate-based synchronization approach [2] (see Eq. (1)), its absence in the basic ConCor algorithm [3] is a weak spot. As an alternative, in [3], the global means are removed from $a(t)$ and $b(t)$, and both signals rescaled with their global standard deviations. If the bitrate sequences were wide-sense stationary, this global treatment would be equivalent to true normalization in the sense of ZNCC. For realistic bitrate sequences, however, this can of course only be a first approximation.

In the following, modifications to the basic ConCor algorithm will be proposed in order to incorporate normalization. To this end, several auxiliary quantities need to be defined. In particular, the following moving averages computed over M consecutive samples of $a(t)$ are required:

$$\overline{a}(\Delta t) = \frac{1}{M} \sum_{t=\Delta t}^{\Delta t+M-1} a(t) \qquad \textit{(moving average)} \quad (2a)$$

$$\overline{a^2}(\Delta t) = \frac{1}{M} \sum_{t=\Delta t}^{\Delta t+M-1} a^2(t) \qquad \textit{(mvg. avg. energy)} \quad (2b)$$

$$\sigma_a(\Delta t) = \sqrt{\overline{a^2}(\Delta t) - \left(\overline{a}(\Delta t)\right)^2} \qquad \textit{(mvg. std. dev.)} \quad (2c)$$

Corresponding measures are considered for the individual segments $b_i(t)$:

$$\overline{b}_i = \frac{1}{M} \sum_{t=(i-1)M}^{iM-1} b(t) \qquad \text{(segment average)} \quad (3a)$$

$$\overline{b_i^2} = \frac{1}{M} \sum_{t=(i-1)M}^{iM-1} b^2(t) \qquad \text{(average segment energy)} \quad (3b)$$

$$\sigma_{b_i} = \sqrt{\overline{b_i^2} - \overline{b}_i^{\,2}} \qquad \text{(segment standard deviation)} \quad (3c)$$

The *normalized partial cross-correlation functions* (NPCCFs), defined as the ZNCC of $a(t)$ with each of the $b_i(t)$, can then be expressed as follows:

$$\widetilde{c}_i(\Delta t) = \frac{1}{M} \sum_{t=(i-1)M}^{iM-1} \frac{a(t+\Delta t) - \overline{a}_i(\Delta t)}{\sigma_{a_i}(\Delta t)} \cdot \frac{b_i(t) - \overline{b}_i}{\sigma_{b_i}}, \quad (4)$$

where $\overline{a}_i(\Delta t) := \overline{a}(\Delta t + (i-1)M)$ and $\sigma_{a_i}(\Delta t) := \sigma_a(\Delta t + (i-1)M)$ are shifted versions of the moving average and standard deviation from Equations (2a) and (2c).

It can be shown that the NPCCFs $\widetilde{c}_i(\Delta t)$ relate to the corresponding PCCFs $c_i(\Delta t)$ from Section IV-A in the following way:

$$\widetilde{c}_i(\Delta t) = \frac{c_i(\Delta t) - M \cdot \overline{a}_i(\Delta t) \cdot \overline{b}_i}{M \cdot \sigma_{a_i}(\Delta t) \cdot \sigma_{b_i}}$$

It follows that, in order to combine several NPCCFs, they need to be denormalized, added up, and the sum renormalized according to the union of all participating segments. Let $I = \{i_1, i_2, \ldots, i_s\}$ be the index set of s segments to be combined. The combination of NPCCFs, denoted by $\widetilde{c}_I(\Delta t)$, can then be

calculated as follows:

$$\widetilde{c_I}(\Delta t) = \frac{\sum_{i\in I}\left[\sigma_{a_i}(\Delta t)\cdot\sigma_{b_i}\cdot\widetilde{c_i}(\Delta t)+\overline{a}_i(\Delta t)\cdot\overline{b}_i\right]-s\cdot\overline{a}_I(\Delta t)\cdot\overline{b}_I}{s\cdot\sigma_{a_I}(\Delta t)\cdot\sigma_{b_I}} \quad (5)$$

The quantities relating to the union of segments can be readily calculated from the corresponding measures initially computed subject to (2a), (2b), (IV-Ba), (IV-Bb):

$$\overline{a}_I(\Delta t) = \frac{1}{s}\sum_{i\in I}\overline{a}_i(\Delta t) = \frac{1}{s}\sum_{i\in I}\overline{a}(\Delta t+(i-1)M) \quad (6a)$$

$$\overline{a_I^2}(\Delta t) = \frac{1}{s}\sum_{i\in I}\overline{a_i^2}(\Delta t) = \frac{1}{s}\sum_{i\in I}\overline{a^2}(\Delta t+(i-1)M) \quad (6b)$$

$$\sigma_{a_I} = \sqrt{\overline{a_I^2}(\Delta t)-\overline{a}_I^2(\Delta t)} \quad (6c)$$

$$\overline{b}_I = \frac{1}{s}\sum_{i\in I}\overline{b}_i\,, \quad \overline{b_I^2} = \frac{1}{s}\sum_{i\in I}\overline{b_i^2}\,, \quad \sigma_{b_I} = \sqrt{\overline{b_I^2}-\overline{b}_I^{\,2}} \quad (7)$$

Ultimately, the basic ConCor algorithm from [3] only requires modification in that additional quantities be precomputed, and steps 3a and 5 be adjusted:

NORMALIZED CONCOR ALGORITHM

1) Chop the shorter signal into $m = \lfloor L_b/M \rfloor$ segments $b_i(t)$ of equal length M.
2) Compute $\overline{a}(\Delta t)$, $\overline{a^2}(\Delta t)$, $\sigma_a(\Delta t)$, and all $\overline{b}_i$, $\overline{b_i^2}$ and σ_{b_i} according to the equations in (2) and (IV-B), as well as the NPCCFs $\widetilde{c_i}(\Delta t)$ following Equation (4).

Repeat

3a) Make a random selection of s NPCCFs and combine them according to Equation (5).
3b) Extract candidate offsets from the combination $\widetilde{c_I}(\Delta t)$.
3c) For every offset candidate, evaluate the number of consenting NPCCFs (inliers).

until confidence is reached that at least one outlier-free NPCCF set has been selected.

4) Select the offset with most consenting NPCCFs.
5) Recompute the offset from the combination of all consenting NPCCFs according to Equation (5).

■

In step 3b, offset candidates are selected as the positions of local maxima in $\tilde{c}_I(\Delta t)$. The consensus check in step 3c is performed by counting the number of NPCCFs $\tilde{c}_i(\Delta t)$ that have local maxima close to the examined candidate offset. More details can be found in [3].

The added computational burden due to normalization is moderate, amounting to $O(1)$ for the precomputations in step 2. In steps 3a and 5 of the algorithm, the NPCCF combination according to Equations (5) through (7) comes at a complexity increase by $O(n)$, based on the input signal lengths.

The tremendous advantage, however, is the improved localizability of segments $b_i(t)$ within the longer sequence $a(t)$, with all the benefits that ZNCC has over unnormalized cross-correlation.

C. Optimal ConCor Segmentation Parameters

There is the fundamental trade-off in choosing a segment length M and the number s of segments to be combined in each RANSAC step. On the one hand, it is desirable to have as many samples as possible contribute to the cross-correlation functions computed in every iteration. On the other hand, increasing M leads to a higher risk of involving corrupt samples in each segment, and raising s obviously increases the chance to include one or more outlier segments. In the following, we will explain how to determine the optimal choice for M and s, maximizing the total number of involved samples while maintaining a high probability that RANSAC can successfully deliver a solution. The event of RANSAC success is defined as follows:

$\mathcal{R}_N$: "An outlier-free set of segments is selected at least once in at most N random draws."

The optimization problem to be solved is then given by

$$\max_{s,M} sM, \qquad \text{s.t.} \quad \Pr\{\mathcal{R}_N\} \geq 0.99, \quad M \geq 50, \quad (8)$$

where an arbitrarily high confidence level of 99% was chosen. The lower bound on M stipulated in (8) is intended to ensure that individual (N)PCCFs are sufficiently meaningful to be used in consensus checks (during step 3c of the algorithm in Section IV-B). The minimum length of 50 samples is empirically motivated and corresponds to 2 seconds of video for 25 fps footage. It should be noted that this constraint on M is hardly determining for the maximization in practical scenarios. The parameter N bounds the complexity of ConCor, with typical values ranging between 1000 and 10000.

The following notation and assumptions will further be used: The sequences $a(t)$ and $b(t)$ are L_a and L_b samples long, where $L_a \geq L_b$ by definition. Consequently, $b(t)$ contains $m = \lfloor L_b/M \rfloor$ full segments of length M, any remnant samples are ignored. We assume a worst-case overlap between $a(t)$ and $b(t)$ of at least L_0 samples ($L_0 \leq L_b$).

As we have seen earlier, there are two main effects that render a particular sample $b(t)$ unapt for synchronization. Either it lies outside the common overlap region of the two signals, or it is part of a burst error in the video content itself (due to occlusions, lack of motion, etc.). We treat both phenomena jointly by postulating a Markovian burst error model and a uniformly distributed offset between the bitrate signals. The burst error model is characterized by B, the mean error burst length, and p, the overall ratio of samples that are part of error bursts. The uniform offset distribution is determined by the minimum overlap L_0. Fig. 3 shows an example realization generated by this error model.

Both B and p can be set to default values or adapted to a particular class of scenarios, allowing for specific values of the expected duration and frequency of disturbances. For instance, videos acquired during a sports event will contain many frequent but possibly brief occlusions (high p, short B)

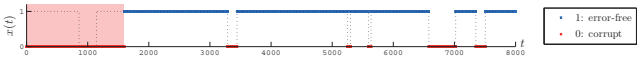
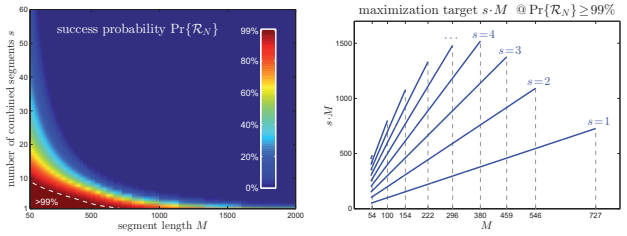


Figure 3. Possible realization of the sample state $x(t)$ of sequence $b(t)$ according to our joint error model, with $L_a = 10000$, $L_b = 8000$, $L_0 = 4000$, $p = 10\%$ and $B = 150$. The dotted line indicates the Markov process generating error bursts, the region shaded in red marks the samples outside the mutual overlap of $a(t)$ and $b(t)$.



(a) (b)

Figure 4. (a) The ConCor success probability for the specific error model from Fig. 3 with $L_a = 10000$, $L_b = 8000$, $L_0 = 4000$, $p = 10\%$, $B = 150$, $N = 10000$. For the values of s and M where $\Pr\{\mathcal{R}_N\} \geq 99\%$, the number of effectively used samples $s \cdot M$ is plotted in (b).

whereas, *e.g.*, surveillance footage is in general steadier (low p, longer B). Machine learning can be used to train these parameters.

In a very abstract form, the probability of success is a monotonically decreasing function in s and M that also depends on the parameters describing the signals and the error model (gathered here in the vector $\mathbf{p}$):

$$\Pr\{\mathcal{R}_N\} = f_{\mathbf{p}}(s, M), \quad \text{with } \mathbf{p} = (L_a, L_b, p, B, L_0, N) \quad (9)$$

In the appendix, we elaborate on the exact relationship between $\Pr\{\mathcal{R}_N\}$ and all the parameters. Here, we only show its qualitative behavior in Fig. 4a. As to be expected, there is a high probability of avoiding corrupt samples when s and M are small, *i.e.*, when only a small number of short segments are used. For increasing values of s and M this probability drops rapidly. There is only a relatively small region in the (s, M) plane for which the success probability exceeds the stipulated 99%.

For all combinations of s and M within this region, Fig. 4b shows the total number of samples $s \cdot M$ that are effectively used in every ConCor iteration. In this example, the global maximum is attained with $s = 4$ and $M = 380$ which corresponds to a total of 1520 samples. An exhaustive search or any suitable discrete maximization strategy can be used to determine these optimal values.

To summarize, we have established a sophisticated probabilistic model that realistically describes the disturbances encountered in bitrate sequences. It links the characteristics of the videos (L_a, L_b), of the expected disturbances (p, B, L_0) as well as the ConCor parameters (s, M, N) to the success probability of our approach. The optimal ConCor segmentation is determined according to (8), demanding maximum signal involvement without sacrificing the confidence in error-free results.

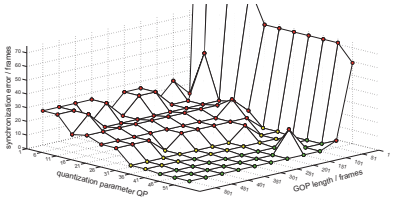


Figure 5. Synchronization error for different parameter settings. Perfect synchronization is indicated by green markers, yellow stands for an absolute error of exactly one frame, red for error values of two frames and above.

V. Practical Considerations regarding Bitrate Extraction

In this section, we propose guidelines for the initial re-encoding process that produces the bitrate profiles as the input for ConCor. We investigate the influence of the two most significant re-encoding parameters on the synchronization performance in Section V-A, and discuss issues and counter measures in case of deficient source material in Section V-B. Finally, the special case where re-encoding is not an option is addressed in Section V-C.

A. Influence of Re-encoding Parameters

In [2], we have proposed to generate bitrate sequences by re-encoding a given video using the H.264/AVC compliant x264 encoder implementation [27]. Obviously, a fixed quantizer needs to be used to produce variable bitrate (VBR) output, and bi-directional prediction disabled in order not to disrupt the sequence's chronology. The most crucial parameters in the re-encoding process are hence the quantization parameter (QP) which adjusts the fidelity of the re-encoded video, and the length of the group of pictures (GOP), *i.e.*, the distance between I-frames separated by a series of P-frames. In Fig. 5, the synchronization error for a representative video pair is displayed subject to different settings of these parameters. It can be observed that frame accurate synchronization can only be consistently achieved in a region of high QP and GOP length values.

To understand why coarse quantization is beneficial, we need to examine the different bitrate components separately. There are basically two complementary types of information contributing to the bitrate output of a hybrid video encoder: data necessary to represent the motion vectors (MV) of every predicted macroblock, and so called texture data (TEX) which comprises the associated residual prediction error and the contribution of individually encoded INTRA macroblocks, both after transform coding and quantization (also see the introductory illustration in Fig. 1b). Intuitively, one associates scene motion primarily with the MV component. From Fig. 6 we see that in the case of fine quantization, *i.e.*, for small QP values, the contribution of the texture component to the total bitrate is predominant. This is due to an overwhelming fraction of INTRA macroblocks which are inherently unrelated to scene motion. Only for large QPs the bitrate for motion vectors can compete with, and even surpass their TEX counterpart. It is further remarkable how the TEX component assumes the shape of the MV bitrate for increasingly coarse quantization,

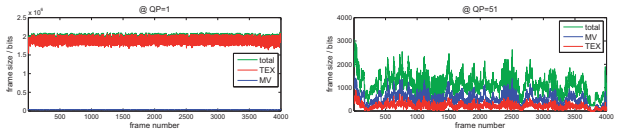


Figure 6. The bitrate profiles for one of the "Human Adam" sequences, at two different quantization parameter values QP and identical GOP length 499 (I frames removed). The TEX and MV components are plotted separately, as well as the total bitrate (TEX+MV+overhead). For QP $= 1$, TEX dominates the total bitrate without following any characteristic evolution. At the other end of the scale, for QP $= 51$, MV comes out on top and imposes its temporal behavior which is closely related to the actual motion present in the video.

and thus also carries information about scene motion. After all, every motion vector is accompanied by a prediction residual contributing to the TEX component. In the case of complex motion, these residuals can be significant and TEX bitrate becomes equally relevant for synchronization. Furthermore, Fig. 7 shows that not only the share of P-type macroblocks increases with coarser quantization but also the fidelity of the associated motion vectors to the actual optical flow describing the scene motion. In accordance with Fig. 5, we propose maximally coarse quantization, with QP values in the 40s and above.

As for a suitable GOP length, it lies in the nature of I-frames to disrupt the temporal dependences in the video stream. Exclusively composed of INTRA macroblocks, they do not carry information about scene motion themselves, yet they are much bigger in size than the relevant P-frames. Consequently, we have proposed to discard and interpolate bitrate values at I-frame positions prior to cross-correlation [2]. Yet we have also found that, especially at coarse quantization, the sudden picture quality refresh at every I-frame affects the size of immediately subsequent P-frames. Depending on the amount and nature of motion in the scene, it takes several frames for this effect to fade away. Consequently, a quickly decaying bitrate peak is observable at the beginning of each GOP. The shorter the chosen GOP length, the more such peaks occur in the re-encoded sequence. In order to mitigate their repercussions on synchronization, significantly long GOPs need to be used. Following our experiences (see also Fig. 5), values beyond 300 frames per GOP are suitable.

B. Re-encoding Artifacts

When bitrate sequences are obtained through re-encoding as described in [2], the nature of the original video data needs to be taken into account. Ideally, the source video is available in hi-quality raw format, ready to be directly encoded into the desired IPPP... scheme, with parameters as suggested in Section V-A. The bitrate samples then coherently reflect the motion complexity throughout the video – except at the I-frame positions which are, for that reason, discarded in our approach [2].

However, more often than not the source videos at hand are compressed in an arbitrary format, and with unknown parameters, both evading direct control. In case of highly compressed source material, we observe the persistence of the original GOP structure after re-encoding. This is due to the typically encountered PSNR differences between the different

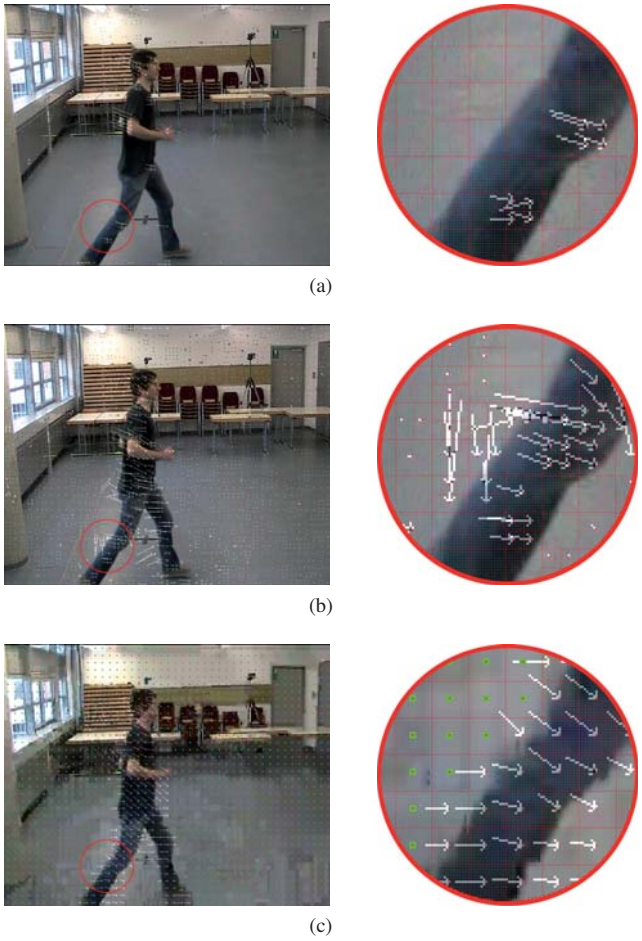


Figure 7. Frame 733 from one of the 'Human Adam' sequences, at quantization parameter values QP $= 1$ (a), 21 (b) and 51 (c), respectively. For fine quantization, large areas of the frame are encoded in INTRA mode, as is evident from the lack of motion vectors in (a). At mid-value QPs, P-type mode is predominant, also in static parts of the frame; the true optical flow is not accurately captured. Only for coarse quantization in (c), most of the static image content is encoded in SKIP mode (indicated by green dots), while the motion vectors exclusively represent the person's movements. Note that in (b) there are multiple motion vectors per macroblock, whereas the majority of motion vectors in (c) represent single macroblocks.

frame types. A former I-frame usually forces the encoder to spend more rate in order to avoid an otherwise disproportionate distortion. This is accomplished by increasing the number of INTRA macroblocks (in case of fine quantization), or by using P-type macroblocks instead of SKIP mode. The same holds for former P-frames which similarly differ in PSNR from possibly present B-frames. The periodicities caused by this combination of persistent GOP structure and overshoots at the beginning of each new GOP (see Section V-A) are of course detrimental during synchronization.

We propose two remedies for this effect. First of all, the periodicities can be diluted by a proper choice of the new GOP length. The periodic length of the spike pattern is given by the least common multiple (lcm) of the GOP lengths G_0 and G_1 before and after re-encoding. If the new GOP length G_1 is chosen to be a prime number, the period always assumes the maximum length of $G_0 \cdot G_1$ samples, irrespective of the actual value of G_0. As a consequence of the enlarged period length, the spike disturbance takes on a more aperiodic character. Con-

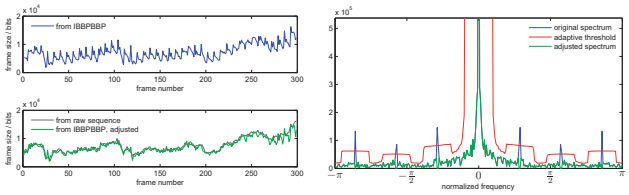
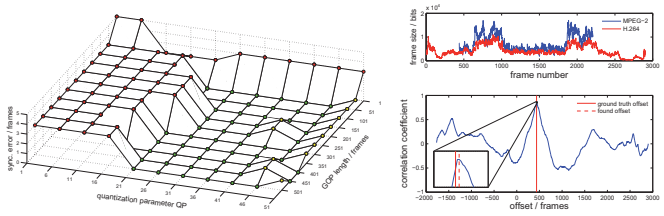


Figure 8. Bitrate sequence extracted (with QP = 40 and GOP length 499) from a low quality video pre-encoded with GOP structure IBBPBBP (top left). The original GOP structure is clearly visible. As reference, the bitrate sequence extracted from the raw image sequence of the same video is given in the bottom left plot. With the adaptive cleanup performed in the Fourier domain, as shown at right, the GOP artifacts can largely be mitigated.



(a) (b)

Figure 9. Bitrate-based synchronization with differing H.264 encoding parameters (a) and outright different codecs (b). In (a), a video (H.264 encoded with QP = 36 and GOP length 451) was synchronized with a second view opposed by 180° and encoded with varying H.264 parameters, leading to the plotted synchronization errors. In (b), bitrate sequences obtained by H.264 (QP = 41, GOP length 499) and MPEG-2 (QP = 31, GOP length 351) are matched with each other. Despite the slight misalignment by 2 frames, the principal temporal similarity is well apparent.

sider the re-encoding of GOPs with $G_0 = 6$ for different values of G_1. With the prime $G_1 = 499$, the spike pattern repeats itself exactly every $\mathrm{lcm}(6, 499) = 6 \cdot 499 = 2994$ samples. Every variation of G_1 would shorten this period drastically, thus emphasizing the periodicity of the spike disturbance. With $G_1 = 500$, $G_1 = 501$ or $G_1 = 498$, for instance, the period length would drop to $\mathrm{lcm}(6, 500) = 1500$, $\mathrm{lcm}(6, 501) = 1002$, or even $\mathrm{lcm}(6, 498) = 498$ samples, respectively.

In addition to choosing a prime GOP length, we propose to actively remove unnaturally strong periodic components from the bitrate sequences after re-encoding. If the re-encoding history of the input videos is known[1], the corresponding frequencies and their harmonics can be precisely suppressed in the Fourier domain. For the case where this information is unavailable, the adaptive spectral cleanup illustrated in Fig. 8 has proven very effective. To this end, all frequencies are examined under a sliding window, and those spectral components set to zero which exceed the average within the window by more than two times the corresponding standard deviation. This approach is versatile enough to cope with other periodicity effects as well, caused by frame rate conversions, for instance, where repeated frames exhibit vanishing rate, and dropped ones disrupt predictability, leading to an elevated rate for the subsequent frame.

C. Synchronization without Re-encoding

In some scenarios it might be impractical to specifically re-encode the source videos. Instead, in order to obtain useful bitrate sequences, the frame sizes are determined by parsing the existing representations. In that case, there is no control whatsoever over the exact nature of the encoding.

As long as both videos are encoded in H.264, there is a good chance that synchronization can, to a certain extent, be successful nonetheless. In the experiment reported in Fig. 9a, two H.264 videos produced with differing encoding parameters are synchronized with our approach. From the second video, a characteristic excerpt has been selected to ensure optimal synchronization in the case of identical parameters. For varying encoding parameters, the development of the synchronization error is then monitored. Within a range of reasonably similar QP and GOP length values, frame exact synchronization is achieved. Misalignment is encountered only if the encoding parameters, especially QP, deviate unduly between the videos.

In Fig. 9b, we go even a step further and mix different codecs. In this example, H.264 and MPEG-2 have been used to produce the bitrate sequences for the synchronization process. With the given settings the resemblant temporal behavior of both signals is evident, and ZNCC retrieves the alignment almost perfectly. Although the number of possible configurations is sheer limitless, this last experiment stresses the general feasibility of bitrate-based synchronization for differently encoded input videos.

VI. Experimental Performance Evaluation

In this section, we experimentally validate our synchronization approach with different video sets that exhibit characteristics typically not handled by state-of-the-art video synchronization algorithms. We also include video sets available from external sources to substantiate the performance of our approach.

In all cases presented in this section, we apply normalized ConCor to bitrate sequences generated with the x264 encoder at QP = 40, with GOP length 499 frames. Furthermore, the adaptive spectral cleanup described in V-B is used. The videos in our database, including those presented here, stem from diverse sources, and vary in codecs, resolution, original encoding parameters, etc. (see Table I on page 11). With that said, the used ConCor parameters have been derived from a very generic error assumption which allows for a mean error burst length of $B = 50$ frames and a sample error rate up to $p = 5\%$. We allow at most $N = 10000$ RANSAC iterations and assume a minimum overlap of 50% of the shorter sequence's length. Unless otherwise available, ground truth offsets have been determined by visual inspection.

A. ConCor Tested for Different Effects

Here, we illustrate the performance of ConCor facing four particularly interesting effects that pose severe challenges to the state-of-the-art:

Wide baselines: Approaches that rely on matching texture features suffer from the limited invariance towards view point

[1] A given video might have been re-encoded multiple times, containing traces of several of the GOP structures used each time.

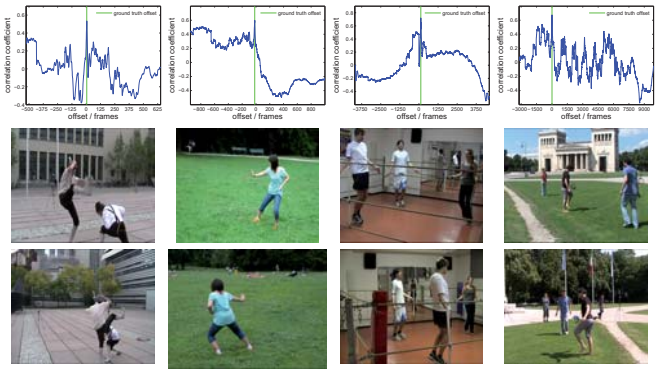


Figure 10. Synchronization results for the *CapoEHA*, *Taiji*, *Rope Skipping* and *Soccer* datasets (from left to right).

changes. Hence feature-based approaches not only exclude scenes where the cameras stand opposite to one another but also scenes with wide angle overlapping views.

Camera motion: A "shaking" camera renders many synchronization algorithms unusable. However, the application of video synchronization of casually captured multi-perspective events requires an algorithm that can handle also this kind of scenario.

Dynamic backgrounds: A changing background can confuse any synchronization algorithm in identifying the real object of interest.

Occlusions: Another problematic effect, especially for approaches that involve the tracking of features, is the temporary disappearance of objects of interest, as well as self-occlusions of multiple moving objects.

In the following we address the issues pointed out above with four synchronization scenarios. The used datasets are presented, together with synchronization results, in Fig. 10, and in Table I. The last two columns in Table I compare the offset found with ConCor to the respective ground truth offset (both in frames). For perfect, frame-accurate synchronization these numbers are identical.

In the *CapoEHA* video pair, the perspectives differ in viewing direction by approximately 90°. Furthermore, since the two subjects wildly dance around each other, frequent self-occlusions occur, and motion blur becomes an issue. The background appears completely different in the two views but is mostly static, with the exception of single pedestrians. One of the cameras is hand-held which introduces slight shaking movements. Frame accurate synchronization is achieved by our approach nonetheless.

The *Taiji* videos have been recorded with different cameras and differ in spatial resolution. This scene contrasts with the previous one in that scene motion is very subtle, whereas background motion is more prominent. The cameras are more than 90° apart, both are hand-held. In one view, several persons walk behind the main actress, in the other one non-involved persons are visible in the background playing ball. Despite these unrelated distractions ConCor successfully syncs the videos and finds the correct offset.

In our third demonstration, three individuals engage in rope skipping. The inherently repetitive motion pattern can be expected to be highly challenging for a method that seeks to find the offset by correlating a motion measure. Nevertheless, ConCor successfully recovers the temporal delay. A remarkable detail about these videos is the wall-filling mirror in the background which obviously aids our approach by multiplying the effective amount of scene motion.

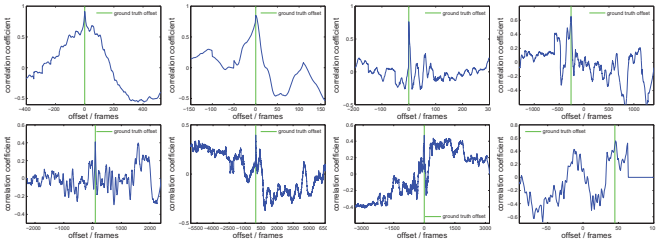


Figure 11. Synchronization outcome for the external datasets (from left to right): *Dog*, *Martial Arts*, *HumanEva*, *Basketball* (top row), *Hall*, *Rothman*, *Magician* and *Train* (bottom row).

Finally, we present a scenario with a very realistic composition of all the discussed effects. In the *Soccer* videos, a ball is kicked back and forth by two individuals who are surrounded by spectators filming the event. The selected views are diametrically opposed to each other and show significant background motion; one of the cameras is hand-held. The particular challenge is that both players regularly leave the cameras' fields of view, which leads to inconsistencies to be detected and eliminated by ConCor. According to the synchronization result, ConCor is able to achieve this, yielding accurate alignment.

B. Evaluation of ConCor on External Data Sets

Here, we demonstrate ConCor's performance using relevant video sets available for download from external sources. We particularly focus on video sets provided by authors of other video synchronization algorithms. Table I and Fig. 11 summarize the videos and the obtained synchronization results. Again, by comparison of the last two table columns the accuracy of our synchronization results can be assessed. Some of the available clips are markedly short, comprising few hundred frames only. An asterisk in the table's third column indicates videos too short to be divided into enough segments of acceptable length. An optimal choice of the s and M parameters according to (8) being impossible in these cases, we select the parameters such that the prospect of success $\Pr\{\mathcal{R}_N\}$ is maximized for the least permissible segment length, *i.e.*, $M=50$ frames and $s=1$. In the case of the extremely short *Train* sequences, it is necessary to further lower the segment length to $M=45$, in order to obtain at least three segments for ConCor to operate on.

The *Dog*, *Martial Arts* and *HumanEva* datasets have all been produced in a meticulously controlled studio environment, and show one or two active performers in front of orderly, static backgrounds. From these sets, video pairs with increasing viewpoint difference, ranging from around 20° to 180°, have been chosen. All of them are successfully synchronized by our approach. A particularity about the selected *Martial Arts* pair is that one of the videos shows a top view of the scene while the second one portrays the action from an unusually low worm's eye perspective.

The *Basketball* and *Hall* sequences have been recorded in natural environments, both with stationary cameras facing each other. They show the outdoor practice of two basketball players and an ante room scenario with multiple persons taking seats and moving on, respectively. The main difficulty lies in the fact that the performers repeatedly enter and leave their scenes, an effect ConCor proves able to cope with. It should be noted that the *Basketball* sequences lack distinctive cues that would allow to conclusively determine the (actually mid-frame) ground truth offset from the interlaced footage. Consequently, the observed synchronization error could liberally be interpreted as amounting to half a frame.

The very challenging *Rothman* and *Magician* datasets show an outdoor street performance and an indoor magic show, respectively. The selected videos are affected by heavy rocking motion and camera pans. In the *Magician* set, the cameras are even displaced by several meters during the recording. In the *Rothman* set, crowds of spectators constitute a rather dynamic background, with motion intensities comparable to those of the street performer himself. A particularly interesting effect in one of the *Magician* videos is a blackout during which the screen remains black for several hundred frames. It is also noteworthy that there is a difference in image format (portrait vs. landscape), a factor to which our synchronization approach is inherently invariant. ConCor sucessfully excludes the blackout and, altogether for both the *Rothman* and *Magician* datasets, retrieves the sequences' offset (up to one frame).

Finally, the *Train* dataset points out the limits of our bitrate based approach. One problem, as mentioned earlier, is the very limited number of available frames ($L_b = 146$). The main challenge, however, is the indistinctive, linear motion described by the depicted toy train, which is also superimposed by relatively strong camera shaking. The only cues useful for synchronization (both for our algorithm and a human observer) are the train's alternately flashing head lights and two hardly discernible collisions with other toys in the scene towards the end of the recordings. Without violating the lower bound on M, stipulated in (8), no more than two segments would be available. But as it turns out, only the last third of the shorter signal is actually suitable for our bitrate based approach. With $M = 45$, three segments are obtained, but each of the corresponding NPCCFs favors a distinct set of potential offsets. Given the dissent among the three NPPCFs, it would be preposterous to speak of a "consensus"-based decision in this case. Which segment eventually becomes classified as *the* inlier is basically random: in one out of three cases this choice is made in favor of the actually faithful third segment. This needs to be borne in mind regarding the rather accurate synchronization result reported here.

VII. Concluding Remarks

With the bitrate-based description of scene changes, in combination with consensus-based cross-correlation, we have presented a reliable and versatile video synchronization algorithm. In comparison with the state-of-the-art, the approach is highly independent of camera, image and scene properties. Owing to the very abstract quantification of relevant scene changes, attributes such as source codec, image resolution, orientation, brightness, etc. have no or very limited influence on the synchronization process. Since the amount of scene changes is quantified, rather than their exact appearance, true viewpoint independence is achieved. Furthermore, the proposed approach mostly compensates for global camera motion, a capability directly inherited from the underlying qualities of H.264/AVC. The general approach is complemented with normalized ConCor which deals with distracting, unrelated object motion, monotonous signal parts, occlusions, sudden camera motion too strong to be fully compensated by H.264/AVC, and other temporary, disturbing effects. The only two requirements of the presented synchronization approach are the presence of meaningful scene changes, usually in the form of object motion, and the continuous focus of both cameras on the same objects of interest. The prototypical application scenario where these requirements are met is multi-view video sharing, where uploaded video clips usually undergo re-encoding into a common format. During this process the bitrate data necessary for synchronization can be tapped at no extra cost. In other scenarios where re-encoding is not an option, one can resort to bitrate profiles extracted from differently encoded videos as well. The full potential of this approach is yet to be explored and part of future research.

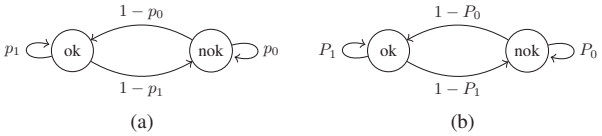


Figure 12. The Markov chains governing the occurrence of corrupt samples (a), and outlier segments (b), respectively.

In contrast to other video synchronization methods, our approach as presented in this article does not yield sub-frame accuracy. With little extra effort, this can be achieved by interpolation of the bitrate sequences at the desired resolution. It remains to be investigated how the performance of this straightforward approach compares to the more sophisticated, and by far more complex, frame rate upconversion of the initial video data.

Appendix
Probabilistic Error Model behind ConCor

Modeling Error Bursts

To realistically model the occurrence of error bursts, a homogeneous, two-state Markov chain is devised with transition matrix $\mathbf{P} = \left(\begin{smallmatrix} p_0 & 1-p_1 \\ 1-p_0 & p_1 \end{smallmatrix}\right)$, as depicted in Fig. 12a. Let $x(t)$ denote the state of sample $b(t)$ which can either be error-free ($x(t) = 1$) or corrupt ($x(t) = 0$) (see also Fig. 3). The transition probabilities $p_0 = \Pr\{x(t)=0 \,|\, x(t-1)=0\}$ and $p_1 = \Pr\{x(t)=1 \,|\, x(t-1)=1\}$ determine how likely it is for the current sample to remain in the same state as the previous one. The complementary probabilities $(1-p_0)$ and $(1-p_1)$ quantify the rate of state changes, accordingly.

In order to describe the Markov chain in a more intuitive way, we parametrize it by its *mean error burst length* B, and its *steady-state sample error rate* p. For an error burst to last l samples, the Markov chain needs to remain in the error state

Table I
SUMMARY OF THE PRESENTED VIDEO SETS

Video Set Name	Video Properties	ConCor Parameters	Ground Truth	Sync. Result
CapoEHA [28] — scene 4 — views 1 & 4	• viewpoint difference: $\sim 90°$ • stationary / hand-held cameras • $L_a = L_b = 934$ • highly dynamic scene, frequent self-occlusions • MPEG-2, 25 fps, 720×576 pixels	$M = 62$ $s = 4$	5	5
Taiji [28] — views 2 & 3	• viewpoint difference: $> 90°$ • hand-held cameras • $L_a = 1184$, $L_b = 1177$ • smooth, subtle motion • WMV-9, 25 fps, 720×576 / 640×480 pixels	$M = 65$ $s = 5$	-18	-18
Rope Skipping [28]	• viewpoint difference: $< 90°$ • stationary cameras • $L_a = 4836$, $L_b = 4584$ • periodic motion, mirror in scene • MPEG-2, 25 fps, 720×576 pixels	$M = 208$ $s = 5$	122	122
Soccer Amateurs [28] — views 1 & 4	• viewpoint difference: $\sim 180°$ • stationary / hand-held cameras • $L_a = 10068$, $L_b = 4401$ • dynamic background, subjects leave visual field • MPEG-2, 25 fps, 720×576 pixels	$M = 153$ $s = 4$	4	4
Dog [29] — scene *Walking* — views 0 & 1	• viewpoint difference: $\sim 20°$ • stationary cameras • $L_a = L_b = 581$ • PNG image sequence, 25 fps, 1624×1080 pixels	$M = 58$ $s = 2$	0	0
Martial Arts [29] — scene *Kick One* — views 0 & 6	• viewpoint difference: $\sim 90°$ (worm's eye vs. top view) • stationary cameras • $L_a = L_b = 211$ • PNG image sequence, 25 fps, 1624×1080 pixels	$M = 50^*$ $s = 1$	0	0
HumanEva [30] — scene *S1-Box1* — views C2 & C3	• viewpoint difference: $\sim 180°$ • stationary cameras • $L_a = L_b = 360$ • MPEG-4, 60 fps, 640×480 pixels	$M = 50^*$ $s = 1$	0	0
Basketball [7]	• viewpoint difference: $\sim 180°$ • stationary cameras • $L_a = 1881$, $L_b = 1798$ • Indeo v5, 25 fps, 720×576 pixels	$M = 119$ $s = 4$	-262	-263
Hall [7]	• viewpoint difference: $< 180°$ • stationary cameras • $L_a = 2638$, $L_b = 2533$ • Indeo v5, 25 fps, 720×576 pixels	$M = 133$ $s = 5$	94	94
Rothman [31] — views 1 & 2	• viewpoint difference: $\sim 90°$ • hand-held cameras • $L_a = L_b = 6899$ • Lagarith, 25 fps, 960×544 / 544×960 pixels	$M = 344$ $s = 4$	0	1
Magician [31] — views 2 & 3	• viewpoint difference: $\sim 45°$ (varying) • hand-held cameras • $L_a = L_b = 3800$ • Lagarith, 25 fps, 960×544 / 544×960 pixels	$M = 120$ $s = 4$	0	-1
Train [15]	• viewpoint difference: $\sim 45°$ • hand-held cameras • $L_a = 200$, $L_b = 146$ • MPEG-4, 25 fps, 720×576 pixels	$M = 45^*$ $s = 1$	45	46

exactly $(l-1)$ times before leaving it. The probability for this to happen is $p_0^{l-1}(1-p_0)$, and consequently:

$$B = \mathrm{E}\{l\} = \sum_{l=1}^{\infty} l\, p_0^{l-1}(1-p_0) = \frac{1}{1-p_0} \tag{10}$$

The stationary distribution of $x(t)$ can be computed as the eigenvector of $\mathbf{P}$ corresponding to its unity eigenvalue. It is thus the solution to $\left(\begin{smallmatrix} p_0-1 & 1-p_1 \\ 1-p_0 & p_1-1 \end{smallmatrix}\right)\left(\begin{smallmatrix} p \\ 1-p \end{smallmatrix}\right) = \mathbf{0}$, resulting in

$$p = \frac{1-p_1}{2-p_0-p_1}. \tag{11}$$

Accordingly, the recurrence probabilities can be expressed in terms of B and p in the following way:

$$p_0 = \frac{B-1}{B}, \qquad p_1 = 1 - \frac{p}{B(1-p)} \tag{12}$$

Given this model for the sample state parametrized by p and B, let's now turn to the error distribution for segments

of length M. There are $m = \lfloor L_b/M \rfloor$ such segments, each of which is considered an outlier if it contains one or more corrupt samples. The random process determining the occurrence of outlier segments then obeys a Markov chain too, very similar to the one governing the sample state (see Fig. 12b). We will use capital letters for the segment-based quantities, namely $X_i \in \{0,1\}$ for the state of the i-th segment $b_i(t)$, and P_0, P_1 and P for the recurrence probabilities and the steady-state outlier rate, respectively.

If the previous segment was an inlier, it takes M recurring error-free samples for the current segment to be entirely error-free as well, thus

$$P_1 = \Pr\{X_i{=}1 | X_{i-1}{=}1\} = p_1^M. \tag{13}$$

An arbitrary segment is an inlier if its first sample is error-free, and the $(M{-}1)$ subsequent samples are too. Accordingly, the complementary event that the segment is an outlier occurs with probability

$$P = 1 - (1-p)\, p_1^{M-1}. \tag{14}$$

With a relationship analog to the one in Equation (11), the remaining transition probability can then be expressed as

$$P_0 = 2 - P_1 - \frac{1-P_I}{P}. \tag{15}$$

To evaluate the overall RANSAC success probability $\Pr\{\mathcal{R}_N\}$ for a given pair of s and M, we first compute the probability $\Pr\{\mathcal{D}\}$ that, in one of the iterations, a random draw of s segments is successful.

$\mathcal{D}$: "s randomly selected segments are error-free."

Ignoring the overlap effect for the moment, let's define the following random variables:

$$G = \sum_{i=1}^{m} X_i, \quad \text{the no. of good, } \textit{i.e.}\text{, error-free seg.s,} \tag{16}$$

$$H = X_1 + X_m, \text{ and} \tag{17}$$

$$K = \sum_{i=2}^{m} X_{i-1} X_i, \text{ the corr. of adjacent seg. states.} \tag{18}$$

It can be shown [32], [33] that the joint probability of G, H, and K is given by

$$\Pr\{G{=}g, H{=}h, K{=}k\} = \binom{2}{h}\binom{g-1}{k}\binom{m-g-1}{g-k-h} \alpha\, \beta^g\, \gamma^h\, \delta^k \tag{19a}$$

$$\text{where } \alpha = (1-2P+P_1P)^{m-1} / (1-P)^{m-2}, \tag{19b}$$

$$\beta = (1-P_1)^2\, P\, (1-P) / (1-2P+P_1P)^2, \tag{19c}$$

$$\gamma = (1-2P+P_1P) / ((1-P)(1-P_1)). \tag{19d}$$

The probability that, among the m segments of $b(t)$, exactly g are error-free is then given by

$$\Pr\{G=g\} = \sum_{h=0}^{2}\sum_{k=0}^{m-1} \Pr\{G{=}g, H{=}h, K{=}k\}. \tag{20}$$

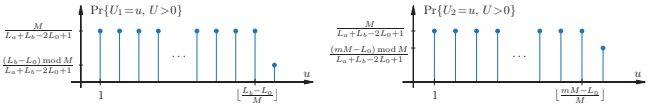


Figure 13. Probability distribution of the numbers U_1 and U_2 of unusable segments at the head and tail of $b(t)$, respectively.

Together with

$$\Pr\{\mathcal{D}|G=g\} = \prod_{i=0}^{s-1} \frac{g-i}{m-i} = \binom{g}{s} \Big/ \binom{m}{s} \tag{21}$$

and

$$\Pr\{\mathcal{R}_N|G=g\} = 1 - (1 - \Pr\{\mathcal{D}|G=g\})^N, \tag{22}$$

we eventually obtain the prospect of success as

$$\Pr\{\mathcal{R}_N\} = \sum_{g=0}^{m} \Pr\{\mathcal{R}_N|G=g\} \Pr\{G=g\}. \tag{23}$$

Incorporating the Overlap Effect

In order to incorporate the so far neglected overlap effect, we introduce the random variable U counting the segments of $b(t)$ that are not fully contained in the overlap region, and thus unusable. Obviously, U depends on the unknown offset between both sequences. If we assume the offset to be uniformly distributed between $(L_0{-}L_b)$ and $(L_a{-}L_0)$, such that the assumed minimum overlap of L_0 samples is guaranteed, there are $(La{+}L_b{-}2L_0{+}1)$ equiprobable shifts in total. For $(L_a{-}mM{+}1)$ of them, all segments of $b(t)$ fully overlap with sequence $a(t)$, hence $U{=}0$. Let's furthermore define U_1 and U_2 as the number of unusable segments at the beginning and at the end of $b(t)$, respectively. By definition, $b(t)$ is the shorter sequence, so either its head *or* its tail protrudes beyond $a(t)$. Hence, the events $U_1{=}u$ and $U_2{=}u$ are mutually exclusive as long as $u > 0$. Accordingly, the distribution of U can be expressed as given in Equation (24) below.

$$\Pr\{U=u\} = \begin{cases} \frac{L_a - mM + 1}{L_a + L_b - 2L_0 + 1} & : u = 0 \\ \Pr\{U_1 = u\} + \Pr\{U_2 = u\} & : u > 0 \end{cases} \tag{24}$$

Shifting the sequence $b(t)$ to the left, more and more segments leave the overlap region, U_1 being incremented every M samples. In the worst case, $\lfloor \frac{L_b - L_0}{M} \rfloor$ segments lie completely outside the overlap region, and another one protrudes by $((L_b - L_0) \bmod M)$ samples. Similarly, if $b(t)$ is shifted to the right, U_2 will increase by one every M samples, eventually leading to $\lfloor \frac{mM - L_0}{M} \rfloor$ non-overlapping segments and one partially protruding by $((mM{-}L_0) \bmod M)$ samples. The resulting distributions of U_1 and U_2 are depicted in Fig. 13.

With U unusable segments due to the overlap effect, the number of available segments in $b(t)$ is de facto reduced from m to $(m{-}U)$. This requires modifications to the equations involved in the computation of $\Pr\{G = g\}$, namely (19a), (19b) and (20), which become

$$\Pr\{G{=}g, H{=}h, K{=}k \,|\, U{=}u\} = \binom{2}{h}\binom{g-1}{k}\binom{m-u-g-1}{g-k-h} \tilde{\alpha}\, \beta^g\, \gamma^h\, \delta^k, \tag{25a}$$

where $\tilde{\alpha} = (1-2P+P_1P)^{m-u-1} \,/\, (1-P)^{m-u-2}$, (25b)

$$\Pr\{G=g\} = \sum_{h,k,u} \Pr\{G=g, H=h, K=k \,|\, U=u\} \Pr\{U=u\}. \quad (26)$$

The rest of the equations, especially (21) through (23), remain valid.

The final result $\Pr\{\mathcal{R}_N\}$ is obtained from (23) using the expressions in (22) and (21), as well as those in (26), (25a), (25b) and (24).

REFERENCES

[1] F. Schweiger, E. Steinbach, M. Fahrmair, and W. Kellerer, "CAMP: A framework for cooperation among mobile prosumers," in *International Conference on Multimedia and Expo*, New York, NY, Jun. 2009.

[2] G. Schroth, F. Schweiger, M. Eichhorn, E. Steinbach, M. Fahrmair, and W. Kellerer, "Video synchronization using bit rate profiles," in *International Conference on Image Processing*, Hong Kong, Sep. 2010, pp. 1549–1552.

[3] F. Schweiger, G. Schroth, M. Eichhorn, E. Steinbach, and M. Fahrmair, "Consensus-based cross-correlation," in *ACM Multimedia*, Scottsdale, AZ, Nov. 2011.

[4] I. Reid and A. Zisserman, "Goal-directed video metrology," *Computer Vision—ECCV'96*, pp. 647–658, 1996.

[5] G. Stein, "Tracking from multiple view points: Self-calibration of space and time," in *Conference on Computer Vision and Pattern Recognition*, Ft. Collins, CO, Jun. 1999.

[6] D. Pooley, M. Brooks, A. Van Den Hengel, and W. Chojnacki, "A voting scheme for estimating the synchrony of moving-camera videos," in *International Conference on Image Processing*, Barcelona, Spain, Sep. 2003.

[7] Y. Caspi, D. Simakov, and M. Irani, "Feature-based sequence-to-sequence matching," *International Journal of Computer Vision*, vol. 68, no. 1, pp. 53–64, 2006.

[8] D. Wedge, D. Huynh, and P. Kovesi, "Motion guided video sequence synchronization," in *Asian Conference on Computer Vision*, Hyderabad, India, Jan. 2006, pp. 832–841.

[9] D. Brito, F. Pádua, R. Carceroni, and G. Pereira, "Synchronizing video cameras with non-overlapping fields of view," in *XXI Brazilian Symp. on Computer Graphics and Image Processing*, 2008, pp. 37–44.

[10] F. Pádua, R. Carceroni, G. Santos, and K. Kutulakos, "Linear sequence-to-sequence alignment," *Trans. on Pattern Analysis and Machine Intelligence*, pp. 304–320, 2010.

[11] A. Whitehead, R. Laganiere, and P. Bose, "Temporal synchronization of video sequences in theory and in practice," in *Workshop on Motion and Video Computing*, vol. 2, 2005, pp. 132–137.

[12] C. Lei and Y. H. Yang, "Tri-focal tensor-based multiple video synchronization with subframe optimization," *Trans. on Image Processing*, vol. 15, no. 9, pp. 2473–2480, Sep. 2006.

[13] C. Rao, A. Gritai, M. Shah, and T. Syeda-Mahmood, "View-invariant alignment and matching of video sequences," in *International Conference on Computer Vision*, Nice, France, Oct. 2003, pp. 939–945.

[14] P. Tresadern and I. Reid, "Synchronizing image sequences of non-rigid objects," in *British Machine Vision Conference*, vol. 2, Norwich, UK, 2003, pp. 629–638.

[15] T. Tuytelaars and L. Van Gool, "Synchronizing video sequences," in *Conference on Computer Vision and Pattern Recognition*, Washington, DC, Jun. 2004.

[16] L. Wolf and A. Zomet, "Correspondence-free synchronization and reconstruction in a non-rigid scene," in *Workshop on Vision and Modelling of Dynamic Scenes, Copenhagen*, 2002.

[17] M. Irani, "Multi-frame optical flow estimation using subspace constraints," in *International Conference on Computer Vision*, vol. 1, 1999, pp. 626–633.

[18] J. Yan and M. Pollefeys, "Video synchronization via space-time interest point distribution," in *Advanced Concepts for Intelligent Vision Systems*, Brussels, Belgium, Sep. 2004.

[19] K. Raguse and C. Heipke, "Photogrammetric synchronization of image sequences," in *ISPRS Commission V Symp. on Image Engineering and Vision Metrology*, Dresden, Germany, Sep. 2006, pp. 254–259.

[20] Y. Caspi and M. Irani, "A step towards sequence-to-sequence alignment," in *Conference on Computer Vision and Pattern Recognition*, Hilton Head, SC, Jun. 2000.

[21] Y. Ukrainitz and M. Irani, "Aligning sequences and actions by maximizing space-time correlations," Graz, Austria, May 2006, pp. 538–550.

[22] Y. Caspi and M. Irani, "Aligning non-overlapping sequences," *International Journal of Computer Vision*, vol. 48, no. 1, pp. 39–51, 2002.

[23] C. Dai, Y. Zheng, and X. Li, "Subframe video synchronization via 3d phase correlation," in *International Conference on Image Processing*, Atlanta, GA, Oct. 2006, pp. 501–504.

[24] M. Ushizaki, T. Okatani, and K. Deguchi, "Video synchronization based on co-occurrence of appearance changes in video sequences," in *International Conference on Pattern Recognition*, Hong Kong, Aug. 2006.

[25] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1, pp. 43–72, 2005.

[26] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H. 264/AVC video coding standard," *Trans. on circuits and systems for video technology*, vol. 13, no. 7, pp. 560–576, 2003.

[27] VideoLAN. x264, the best H.264/AVC encoder. Accessed: Sept. 2011. [Online]. Available: www.videolan.org/developers/x264.html

[28] F. Schweiger. (2012, Apr.) TU München - Lehrstuhl für Medientechnik. [Online]. Available: http://www.lmt.ei.tum.de/florian/sync/#data

[29] G. Janssens. (2009, Dec.) 4d-repository :: Public. [Online]. Available: http://4drepository.inrialpes.fr/public/datasets

[30] L. Sigal, A. Balan, and M. Black, "Humaneva: Synchronized video and motion capture dataset and baseline algorithm for evaluation of articulated human motion," *International Journal of Computer Vision*, vol. 87, no. 1, pp. 4–27, 2010.

[31] L. Ballan, G. Brostow, J. Puwein, and M. Pollefeys, "Unstructured video-based rendering: Interactive exploration of casually captured videos," *ACM Trans. on Graphics*, vol. 29, no. 4, p. 87, 2010.

[32] J. Klotz, "Markov chain clustering of births by sex," *Berkeley Symp. on Mathematical Statistics and Probability*, vol. 4, pp. 173–185, 1972.

[33] ——, "Statistical inference in Bernoulli trials with dependence," *The Annals of Statistics*, vol. 1, no. 2, pp. 373–379, 1973.

Florian Schweiger is a Ph.D. candidate at the Institute for Media Technology. He received his B.Sc. degree in Electrical Engineering and Information Technology from Technische Universität München (TUM), Germany in 2004, and diploma degrees both from TUM and Télécom Bretagne, France in 2007. His research interests cover video synchronization and image feature detection.

Georg Schroth is a Ph.D. candidate at the Institute for Media Technology at Technische Universität München (TUM). He holds a Dipl.-Ing. degree (2008) in Electrical Engineering and Information Technology from TUM. As a Graduate Visiting Researcher at Stanford University he joined the GPS Laboratory in 2007 and the Information Systems Laboratory in 2010. His research focuses on vision-based localization methods.

Michael Eichhorn received his Dipl.-Ing. and Ph.D. degrees from Technische Universität München in 2005 and 2011, respectively. During his time at the Institute for Media Technology, his research interests were in the field of automotive in-car multimedia applications, with focus on video compression. He is now a senior researcher and development engineer with Hexagon Technology in Heerbrugg, Switzerland, working on image-based metrology.

Anas Al-Nuaimi is a Ph.D. candidate at the Institute for Media Technology. He holds a B.Sc. degree in Electrical and Computer Engineering from the Hashemite University in Jordan, and a M.Sc. degree in Communications Engineering from Technische Universität München, Germany. His main research interests lie in the areas of sensor and data fusion, and computer vision in the context of cooperative mobile media.

Burak Cizmeci is a Ph.D. candidate at the Institute for Media Technology. He holds B.Sc. degrees in Electronics Engineering and in Computer Engineering, and a M.Sc. degree from Isik University, Istanbul, Turkey. Currently, he is working on multimodal multiplexing of audio, video and haptic signals for telepresence and teleaction systems.

Michael Fahrmair was awarded Dipl.-Ing. (M.Sc.) and Dr. (Ph.D.) degrees in computer science by the Technische Universität München, Germany, in 1999 and 2005 respectively. He joined DOCOMO Communications Laboratories Europe in 2006 to work in the Ubiquitous Networking group. He is currently working as a Research Manager in the Service Research Group. His main interests are ubiquitous mobile service platforms and rich mobile multimedia communication including 3D video processing, image processing and mixed reality.

Eckehard Steinbach (Senior Member, IEEE) studied electrical engineering at the University of Karlsruhe, Karlsruhe, Germany, the University of Essex, Essex, U.K., and ESIEE, Paris, France. He received the Engineering Doctorate from the University of Erlangen-Nuremberg, Germany, in 1999. From 1994 to 2000, he was a member of the research staff of the Image Communication Group, University of Erlangen-Nuremberg. From February 2000 to December 2001, he was a Postdoctoral Fellow with the Information Systems Laboratory, Stanford University, Stanford, CA. In February 2002, he joined the Department of Electrical Engineering and Information Technology, Technische Universitaet Muenchen, Munich, Germany, where he is currently a Full Professor for Media Technology. His research interests are in the area of audiovisual-haptic information processing and communication as well as networked and interactive multimedia systems.
Dr. Steinbach has been serving as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since 2006 and for the IEEE TRANSACTIONS ON MULTIMEDIA since 2011.