Temporal Matching Kernel with Explicit Feature Maps

Sébastien Poullot, Shunsuke Tsukatani, Anh Phuong Nguyen, Hervé Jégou, Shin'Ichi Satoh

Temporal Matching Kernel with Explicit Feature Maps

Sébastien Poullot
National Institute of Informatics and JFLI (CNRS)
Tokyo, Japan
spoullot@nii.ac.jp

Shunsuke Tsukatani
University of Tokyo and National Institute of Informatics
Tokyo, Japan
tsukatani@nii.ac.jp

Anh Phuong Nguyen
MMLab - University of Information Technology
Ho Chi Minh, Vietnam
anhnp@uit.edu.vn

Hervé Jégou
Inria
Rennes, France
herve.jegou@inria.fr

Shin'Ichi Satoh
National Institute of Informatics and University of Tokyo
Tokyo, Japan
satoh@nii.ac.jp

ABSTRACT

This paper proposes a framework for content-based video retrieval that addresses various tasks as particular event retrieval, copy detection or video synchronization. Given a video query, the method is able to efficiently retrieve, from a large collection, similar video events or near-duplicates with temporarily consistent excerpts. As a byproduct of the representation, it provides a precise temporal alignment of the query and the detected video excerpts.

Our method converts a series of frame descriptors into a single visual-temporal descriptor, called a temporal invariant match kernel. This representation takes into account the relative positions of the visual frames: the frame descriptors are jointly encoded with their timestamps. When matching two videos, the method produces a score function for all possible relative timestamps, which is maximized to obtain both the similarity score and the relative time offset.

Then, we propose two complementary contributions to further improve the detection and localization performance. The first is a novel query expansion method that takes advantage of the joint descriptor/timestamp representation to automatically align the first result set and produce an enriched temporal query. In contrast to other query expansion methods proposed for videos, it preserves the localization capability. Second, we improve the localization trade-off between quality and representation size by using several complementary temporal match kernels.

We evaluate our approach on benchmarks for particular event retrieval, copy detection and video synchronization. Our experiments show that our approach achieve excellent detection and localization results.

1. INTRODUCTION

THIS paper addresses the problem of content-based search in large collections of video clips. Our goal is twofold: (i) identify the content of interest (ii) localize the content of interest within the video clip. We essentially focus on tasks such as particular event retrieval [19], copy detection [15, 25, 7] and video synchronization, for which the temporal consistency is important. Because we want to handle large datasets, the complexity is regarded as a key factor, both in terms of CPU and memory. This disqualifies the numerous approaches that store local image descriptors or spatial-temporal points [15], which are too costly for big datasets.

In contrast, following some recent works for particular event retrieval and video copy detection [19, 8], the input of our approach is a set of compact frame *vectors* produced from local descriptors. A visual frame descriptor is obtained by first encoding the local descriptors of the image with the Fisher vector [16, 17] or another alternative coding approach [24, 12, 2, 14], such as those recently proposed for image classification or search tasks. These vectors are then projected to a few hundreds or thousands components [13, 12] with dimensionality reduction.

With such frame vectors as the initial video representation, the most common options are the following ones:

1. adopt a "bag of frames" shift-invariant representation [8], whose major advantage is to be computationally cheap. The main drawback is that this class of method looses the temporal consistency of the frames, therefore the method is not able to provide the temporal alignment

2. compare videos by matching all the frames of the query video with all the frames of each database video. The potential matching frames are then fed to a Hough temporal matching ensuring that the matches are temporally consistent [7]. However, the complexity of this strategy is quadratic in the length of video clips and is therefore very costly.

To alleviate the quadratic term, the circulant temporal encoding (CTE) of Revaud *et al.* [19] exploits a simple assumption (time translation) on the temporal model to evaluate efficiently, *i.e.*, in the Fourier domain, the cross-correlation of two series of vectors representing two videos. This strategy is inspired by recent works by Henriques *et al.* for efficient detection [10] and tracking [11], who obtained top performances in the VOT2014 object tracking challenge [9]. This approach has the capability of finding the near-duplicates and provides the temporal offset between the videos. Indeed, CTE considers the temporal shift-invariant matching

MM'15, October 26–30, 2015, Brisbane, Australia.
 ISBN 978-1-4503-3459-4/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2733373.2806228.

as matching sequences with all possible shifts, then regards matching sequences with all possible shifts as a convolution. However, this strategy has several shortcomings, the main one being that this method is tailored to a measure of cross-correlation, which is a strong assumption on the time model best fitted to compare videos.

We mention that video recognition based with CTE is outperformed by the recent stable hyper-pooling approach [8]. However hyper-pooling is more costly and, as mentioned previously, does not provide any localization capability, such as the estimation of the temporal offset.

In this paper, we propose a class of *temporal match kernels* which are able to match and align videos efficiently. It can be seen as a generalization of CTE, yet compared to this representation it offers two advantages: (i) the design of the kernel offers degrees of freedom likely to better suit to the particular recognition task ; (ii) the representation has a fixed size and is independent from the video length.

The idea is to encode a set of vectors (the frame descriptors associated with a given video) into a single vector so that the comparison of two videos is performed with dot products, in the spirit of the kernel descriptors of Bo [3], but here specifically adapted to the case of time vector sequences. The design of this encoding strategy, amenable to incorporate the shift-invariance and the capability of finding the temporal offset, relies on explicit feature maps [23]. Compared to kernel descriptors, the main advantage is that we do not have a single score. Instead the comparison outputs a score function for all time-shift hypotheses, in the form of a trigonometric polynomial.

Our paper makes the following contributions:

- We adapt the kernel descriptor framework of Bo [3] to sequences of frames. More specifically, we propose a class of shift-invariant temporal match kernels by gathering the formalism of explicit feature maps [23] and circulant encoding [10, 19]. This strategy is a generalization of CTE, which gives more room to adapt the properties of the matching kernel and limits the approximation artifacts inherent to CTE.
- We propose a query expansion (QE) technique that automatically aligns the videos deemed relevant for the query. The new query is constructed in the Fourier domain and does not require the original frame descriptors. Most importantly and as opposed to possible concurrent QE techniques proposed with compact representations, our method preserves the localization capability of our temporal match kernel.
- We propose a strategy to combine several match kernels into a single one to further improve the trade-off between memory and localization accuracy.

The paper is organized as follows. Section 2 introduces the notation and some preliminary works from which we derive our approach. Section 3 introduces the temporal matching kernels, which is extended to multiple periods to improve the localization in Section 5. Section 4 introduces an original query expansion method for improving the accuracy of the method. The experiments described in Section 6 demonstrate the interest of our approach on public benchmarks for copy detection and particular event retrieval.

2. PRELIMINARIES: FEATURE MAPS

This section briefly introduces a key ingredient of our method, namely the explicit feature maps [23]. It was originally introduced to produce an approximation of a shift-invariant kernel with an explicit mapping, that is, such that the inner product in the embedded space approximates the kernel in the original space. In our case, we employ it to define a class of temporal kernel between set of visual frames.

2.1 Embedding of the kernel function

The idea is to approximate a non-linear kernel $k(\cdot,\cdot) : \mathbb{R}\times\mathbb{R}\to\mathbb{R}$ by using a mapping function $\boldsymbol{\varphi} : \mathbb{R}\to\mathbb{R}^{2m+1}$ as

$$k(u,v) \approx \langle \boldsymbol{\varphi}(u) | \boldsymbol{\varphi}(v) \rangle. \tag{1}$$

This is done for shift-invariant kernels, *i.e.*, such that the kernel can be expressed as $k(u,v) = g(u-v)$, by first considering the Fourier approximation of g as

$$g(z) = \sum_{i=0}^{m} a_i \cos\left(\frac{2\pi}{T} i z\right), \tag{2}$$

which approximates g assuming that it is a T-periodic function. Now, let us consider that $z = u - v$. By employing the elementary trigonometric property

$$\cos(u-v) = \cos u \cos v + \sin u \sin v \tag{3}$$

$$= \left\langle \begin{bmatrix}\cos u\\ \sin u\end{bmatrix} \middle| \begin{bmatrix}\cos v\\ \sin v\end{bmatrix} \right\rangle, \tag{4}$$

and further assuming that the coefficients of the Fourier series are all positive, *i.e.*, $\forall i\ a_i > 0$, the approximation (2) is written as

$$g(u-v) \approx \underbrace{\begin{bmatrix} \sqrt{a_0} \\ \sqrt{a_1}\cos\left(\frac{2\pi}{T}u\right) \\ \sqrt{a_1}\sin\left(\frac{2\pi}{T}u\right) \\ \vdots \\ \sqrt{a_m}\cos\left(\frac{2\pi}{T}mu\right) \\ \sqrt{a_m}\sin\left(\frac{2\pi}{T}mu\right) \end{bmatrix}^\top}_{\boldsymbol{\varphi}(u)^\top} \underbrace{\begin{bmatrix} \sqrt{a_0} \\ \sqrt{a_1}\cos\left(\frac{2\pi}{T}v\right) \\ \sqrt{a_1}\sin\left(\frac{2\pi}{T}v\right) \\ \vdots \\ \sqrt{a_m}\cos\left(\frac{2\pi}{T}mv\right) \\ \sqrt{a_m}\sin\left(\frac{2\pi}{T}mv\right) \end{bmatrix}}_{\boldsymbol{\varphi}(v)}, \tag{5}$$

where the interesting property is that $\boldsymbol{\varphi}(u)$ is independent of v. Therefore, it is possible to construct the mapping $\boldsymbol{\varphi}$ to approximate the shift-invariant kernel as proposed in (1). The approximation quality depends on the quality of the approximation of k by Fourier series [23].

2.2 Application to frame time stamps

Considering two videos V_x and V_y that are temporally perfectly aligned, the distance between their time stamps is $t_x - t_y = 0$. In this ideal case, the distance is coded by a Dirac delta function:

$$\begin{aligned} \text{if } t_x = t_y \quad & \text{then } \mathcal{D}(t_x,t_y) = 1 \\ & \text{else } \mathcal{D}(t_x,t_y) = 0 \end{aligned} \tag{6}$$

The Fourier transform of a Dirac is a constant function, meaning that all a_i are equal in (2). Using the corresponding embedding function is what implicitly done in the circulant encoding technique of Revaud *et al.* [19]. The Dirac's approximation is the best Fourier expansion, w.r.t. square loss, when the function is evaluated at certain positions $(2\pi i z/T,\ i = 0 \dots m-1)$. However, this choice has two drawbacks in our application scenario:

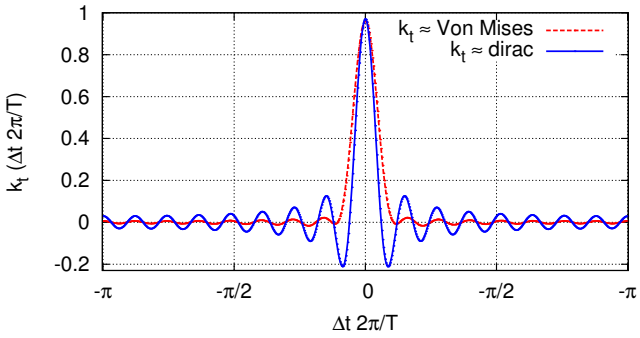

Figure 1: Approximate Dirac and Von Mises: response as a function of the temporal difference when approximating these kernels with finite Fourier series ($m = 16$ frequencies here). As one can observe, the Dirac approximation [19] suffers from oscillations.

1. To get a compact representation, we approximate the temporal kernel with a limited number of frequencies, that is, much smaller that the number of frames to be encoded. The Fourier series we consider are truncated and therefore the approximation obtained when keeping the first components is noisy, suffering strong oscillations around 0, as shown by Figure 1.

2. We want to offer tolerance in time for matching frame by designing a specific shape of the match kernel.

For these reasons, we consider other choices, like the modified Von Mises distribution, a.k.a. the periodic Gaussian, and considered in recent papers [3, 21]. Its Fourier expansion is determined in a close-form manner by the Fourier coefficients

$$a_0 = \frac{B_0(\beta) - e^{-\beta}}{2\sinh(\beta)} \qquad a_i = \frac{B_i(\beta)}{\sinh(\beta)} \text{ for } i \geq 1 \quad (7)$$

where $B_n(\beta)$ is the modified Bessel function of the first kind of order n. Figure 1 clearly shows that the approximate Von Mises temporal kernel gives a flat response out of the target bandwidth, meaning that there are significantly less interferences in the match kernel than with the circulant temporal encoding technique [19].

3. TEMPORAL MATCH KERNELS

This section introduces the class of match kernels proposed in this paper. We first detail the class of kernels that we want to approximate and then show how to approximate them with feature maps encoding. Finally, we show that we can decode them efficiently for any temporal shift. For notation, we denote in bold vectors, tuples and sequences.

3.1 Temporal match kernels with feature maps

Modulating descriptors with timestamps. We consider that each image/frame $\boldsymbol{f_x}$ is associated with a tuple $(\boldsymbol{x}, t_x)$, where $\boldsymbol{x} \in \mathbb{R}^d$ is a d-dimensional vector and t_x is a scalar timestamp. We define a kernel between such tuples that is of the form:

$$k\left(\boldsymbol{f_x}, \boldsymbol{f_y}\right) = k_{\mathrm{f}}(\boldsymbol{x}, \boldsymbol{y})\, k_{\mathrm{t}}(t_x, t_y) \quad (8)$$
$$= \langle \boldsymbol{x} | \boldsymbol{y} \rangle\, k_{\mathrm{t}}(t_x, t_y) \quad (9)$$
$$\approx \langle \boldsymbol{x} | \boldsymbol{y} \rangle\, \boldsymbol{\varphi}(t_x)^\top \boldsymbol{\varphi}(t_y). \quad (10)$$

where we assume that the frame descriptors are normalized and compared with cosine similarity, *i.e.*, $k_{\mathrm{f}}(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{x} | \boldsymbol{y} \rangle$. This kernel is further linearized as

$$k\left(\boldsymbol{f_x}, \boldsymbol{f_y}\right) \approx \langle \boldsymbol{x} \otimes \boldsymbol{\varphi}(t_x) | \boldsymbol{y} \otimes \boldsymbol{\varphi}(t_y) \rangle, \quad (11)$$

where $\otimes$ denotes the Kronecker product. In this case, we stress that the tuple $(\boldsymbol{x}, t_x)$ is represented by a single vector $\boldsymbol{x} \otimes \boldsymbol{\varphi}(t_x)$, that serves as the frame descriptor, and which is compared with the inner product. Another way to interpret this joint encoding is to view $\boldsymbol{x}$ as being *modulated* by the timestamp t_x in a continuous manner, so that only frames with the same timestamp can receive a significant contribution: $|k(\mathbf{x}, \mathbf{y})|$ is bounded by $|k_{\mathrm{t}}(t_x, t_y)|$.

Temporal match kernels with feature maps. The expansion in (11) is inspired by the kernel descriptors of Bo *et al.* [3], however it is here based on explicit feature maps instead [23] of efficient match kernels [4]. Following this work, we now consider the context of match kernels. In our case, the goal is to compare two sequences of descriptors $\mathbf{x} = (\boldsymbol{x}_0, \dots, \boldsymbol{x}_t, \dots)$ and $\mathbf{y} = (\boldsymbol{y}_0, \dots, \boldsymbol{y}_{t'}, \dots)$, where the indices implicitly encode the timestamps[1]. We aim at approximating a kernel on sequences of the form

$$\mathcal{K}_0(\mathbf{x}, \mathbf{y}) = \alpha(\mathbf{x})\alpha(\mathbf{y}) \sum_t k_{\mathrm{f}}(\boldsymbol{x}_t, \boldsymbol{y}_t) \quad (12)$$

$$\propto \sum_{t=0}^{\infty} \boldsymbol{x}_t^\top \boldsymbol{y}_t, \quad (13)$$

where the proportionality factors $\alpha(\cdot)$ are such that $\mathcal{K}_0(\mathbf{x}, \mathbf{x}) = \mathcal{K}_0(\mathbf{y}, \mathbf{y}) = 1$. By convention, we also assume that $\boldsymbol{x}_\tau = \mathbf{0}$ $(\tau > t)$ if the sequence is shorter that t. This kernel is equivalent to

$$\mathcal{K}_0(\mathbf{x}, \mathbf{y}) \propto \sum_{t=0}^{\infty} \sum_{t'=0}^{\infty} \boldsymbol{x}_t^\top \boldsymbol{y}_{t'}\, \delta(t, t'), \quad (14)$$

where $\delta(t, t') = 1$ if $t = t'$, otherwise $\delta(t, t') = 0$. Replacing $\delta(\cdot, \cdot)$ by any temporal kernel $k_{\mathrm{t}}(\cdot, \cdot)$ shows that the circulant temporal embedding [19] is a particular case of our more general formulation, corresponding to the case $k_{\mathrm{t}} = \delta$. Using (11), the kernel is then factorized as

$$\mathcal{K}_0(\mathbf{x}, \mathbf{y}) \propto \Big(\underbrace{\sum_{t=0}^{\infty} \boldsymbol{x}_t \otimes \boldsymbol{\varphi}(t)}_{\boldsymbol{\psi}_0(\mathbf{x})} \Big)^\top \Big(\underbrace{\sum_{t'=0}^{\infty} \boldsymbol{y}_{t'} \otimes \boldsymbol{\varphi}(t')}_{\boldsymbol{\psi}_0(\mathbf{y})} \Big), \quad (15)$$

where $\boldsymbol{\psi}_0(\mathbf{x})$ is the vector representation of the sequence $\mathbf{x}$. The video representation is a $(1+2m) \times d$ dimensions matrix, where m is the number of coefficient kept from the Fourier series (a_1 to a_m) and d is the initial dimensionality of the frame vectors. It can also be written as

$$\boldsymbol{\psi}_0(\mathbf{x}) = [\mathbf{V}_0{}^\top, \mathbf{V}_{1,c}{}^\top, \mathbf{V}_{1,s}{}^\top, ..., \mathbf{V}_{m,c}{}^\top, \mathbf{V}_{m,s}{}^\top]^\top \quad (16)$$

where:

$$\mathbf{V}_0 = a_0 \sum_{t=0}^{\infty} \boldsymbol{x}_t \quad (17)$$

$$\mathbf{V}_{i,c} = a_i \sum_{t=0}^{\infty} \boldsymbol{x}_t \cos\left(\frac{2\pi}{T} t\right)$$

$$\mathbf{V}_{i,s} = a_i \sum_{t=0}^{\infty} \boldsymbol{x}_t \sin\left(\frac{2\pi}{T} t\right)$$

[1]The timestamps are not necessarily integers, any real value is possible. We use integers for the sake of simplicity.

Regularization. The temporal linearity in videos induces a strong self-similarity, and results in interferences when matching two similar videos. Such noise degrades the alignment between two videos. In order to limit this undesirable effect we regularize the kernel descriptors to limit the interferences due to self-similarity in videos. Derived from Winener-Khinchin Theorem, the regularization is done by updating $\psi_0(x)$ as follows:

$$\psi_0(x) := \frac{\psi_0(x)}{\sqrt{\frac{1}{m}\sum_{i=1}^{N}(\mathbf{V}_{i,c}^2 + \mathbf{V}_{i,s}^2)}} \tag{18}$$

This regularization is close to the one introduced in [8]. The main difference is that our regularization is symmetric, applied to both query and database descriptors. In what follows, we use the regularized version for better performance.

3.2 Matching with different timestamps

The temporal similarity in (13) assumes that the sequences are temporally aligned, which prevents from retrieving excerpts that are not synchronous with the query video. We now consider the evaluation of a kernel parametrized by a latent variable Δ controlling the relative offset of two sequences $\mathbf{x}$ and $\mathbf{y}$. The match kernel definition becomes

$$\mathcal{K}_\Delta(\mathbf{x},\mathbf{y}) \propto \sum_{t=0}^{\infty} \boldsymbol{x}_t^\top \boldsymbol{y}_{t+\Delta}, \tag{19}$$

which, similar to (15), is approximated by

$$\mathcal{K}_\Delta(\mathbf{x},\mathbf{y}) \propto \Big(\sum_{t=0}^{\infty} \boldsymbol{x}_t \otimes \boldsymbol{\varphi}(t)\Big)^\top \underbrace{\Big(\sum_{t'=0}^{\infty} \boldsymbol{y}_{t'} \otimes \boldsymbol{\varphi}(t'+\Delta)\Big)}_{\boldsymbol{\psi}_\Delta(\mathbf{y})}, \tag{20}$$

where $\boldsymbol{\psi}_\Delta(\mathbf{y})$ indicates that the timestamps of the sequence $\mathbf{y}$ are incremented by the constant Δ. Note that the vector $\boldsymbol{\psi}_\Delta(\mathbf{y})$ corresponds to a translation in the Fourier domain and is therefore obtained from $\boldsymbol{\psi}_0(\mathbf{y})$ by incrementing the phase by the quantity Δ. However, we don't need $\boldsymbol{\psi}_\Delta(\mathbf{y})$ to compute the similarity of the two sequences for any relative time-shift Δ. Instead, similar to Tolias *et al.* [21], we exploit the fact that $\mathcal{K}_\Delta(\mathbf{x},\mathbf{y})$ is a trigonometric polynomial in Δ, denoted by $\mathcal{K}_{\mathbf{x},\mathbf{y}}(\Delta)$ for clarity, when $\mathbf{x}$ and $\mathbf{y}$ are fixed.

The comparison of two videos proceeds in two steps: (i) computation of the coefficients of the polynomial, (ii) evaluation of the score at different frame-shifts. Using equations (16) and (17) this trigonometric polynomial of video descriptors $\mathbf{V}(\mathbf{x})$ and $\breve{\mathbf{V}}(\mathbf{y})$ is written as

$$\begin{aligned}\mathcal{K}_{\mathbf{x},\mathbf{y}}(\theta) = \langle \mathbf{V}_0 | \breve{\mathbf{V}}_0 \rangle \\ + \sum_{n=1}^{N} \cos(n\Delta)(\langle \mathbf{V}_{n,c} | \breve{\mathbf{V}}_{n,c} \rangle + \langle \mathbf{V}_{n,s} | \breve{\mathbf{V}}_{n,s} \rangle) \\ + \sum_{n=1}^{N} \sin(n\Delta)(-\langle \mathbf{V}_{n,c} | \breve{\mathbf{V}}_{n,s} \rangle + \langle \mathbf{V}_{n,s} | \breve{\mathbf{V}}_{n,c} \rangle)\end{aligned} \tag{21}$$

The coefficients of this polynomial are obtained at the cost of $(1+2m)\times d$ elementary operations, *i.e.*, about twice the cost of an inner product in $\mathbb{R}^d$.

Evaluating the values taken by this score function for all valid frame-shifts Δ has a negligible cost. Figure 2 illustrates the trigonometric polynomials of scores obtained for matching and non-matching sequences. The max peak gives the score $S_{\mathbf{xy}} = \max_\Delta \mathcal{K}_{\mathbf{x},\mathbf{y}}(\Delta)$ and the time offset $\delta_{\mathbf{xy}} = \arg\max_\Delta \mathcal{K}_{\mathbf{x},\mathbf{y}}(\Delta)$ between the sequences $\mathbf{x}$ and $\mathbf{y}$. For each query, the candidates in the database are ordered by decreasing scores.

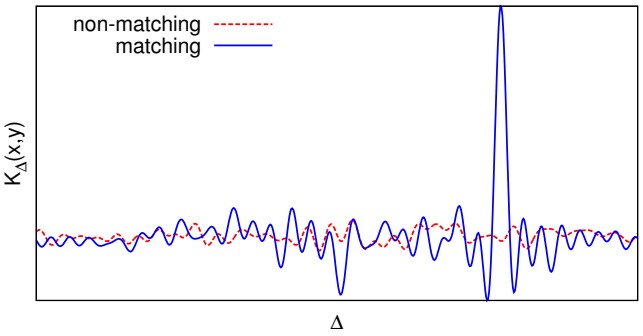

Figure 2: Polynomial of scores $\mathcal{K}_\Delta(\mathbf{x},\mathbf{y})$ for a matching pair of sequences $(\mathbf{x},\mathbf{y})$ (blue) and two unrelated sequences (red).

4. TEMPORAL QUERY EXPANSION

Query expansion (QE), a technique initially introduced in text, is an effective mechanism that improves the recall when matching documents can be reliably identified to build an augmented query. In particular object retrieval, it is commonly combined with spatial re-ranking [6, 18, 5]. The most popular is average query expansion (AQE), which constructs the augmented query by averaging the vectors associated with the images deemed positive after spatial verification is applied on an initial short-list. Other variants were recently proposed, such as discriminative query expansion [1].

4.1 Triplet time consistency

In our case, we consider short vector representations of frames that are incompatible with spatial verification based on local descriptors [5, 22]. Therefore we propose a strategy to first check whether the videos returned in the short-list are temporally consistent or not.

Consider a query $\mathbf{q}$ and a database video $\mathbf{x}$, our temporal match kernel returns a temporal offset $\delta_{\mathbf{qx}}$. Within the short-list $\mathcal{N}_1$ of length $|\mathcal{N}_1|$ returned by the ranking strategy, we check the temporal consistency of each result by considering triplets of offsets. The offsets $\delta_{\mathbf{qx}}$ and $\delta_{\mathbf{qy}}$ between the query $\mathbf{q}$ and two videos $\mathbf{x}$ and $\mathbf{y}$ are byproducts of the initial ranking stage. Additionally, we compute the extra offsets $\delta_{\mathbf{xy}}$ between all pairs $(\mathbf{x},\mathbf{y})$ of videos in the short-list of candidates. These offsets are requested to satisfy the constraint

$$|\delta_{\mathbf{qx}} + \delta_{\mathbf{xy}} + \delta_{\mathbf{yq}}| \leq \varepsilon, \tag{22}$$

where ε is a temporal tolerance related to T. It should be small enough to filter inconsistent matches ($\varepsilon \gg T$) but large enough to tolerate small errors on the estimated offsets. The method is not sensitive to the exact tolerance value. We set $\varepsilon = 100$ frames. Having computed the consistency of each couple of candidates ($(|\mathcal{N}_1|-1)/2$ in total), we produce a per-video consistency measure to partition the short-list as

$$\mathcal{N}_1 = \mathcal{N}_1^{\mathrm{C}} \cup \mathcal{N}_1^{\mathrm{U}}, \tag{23}$$

where $\mathcal{N}_1^{\mathrm{C}}$ is the set of consistent candidates and its supplementary set $\mathcal{N}_1^{\mathrm{U}}$.

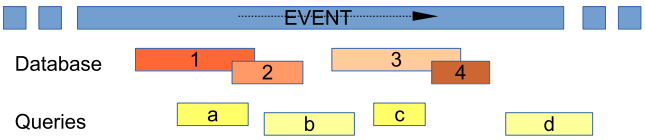

Figure 3: This figure shows how query expansion can retrieve database video clips originally not reachable: in the database video clip **4** is not temporally consistent (and most likely not visually) with any query, however if query **c** returns the database video clip **3** and a new query is generated, based on **c** and **3** then **4** may be reached.

Temporally aligned QE. As discussed in Section 3.2, we compute the descriptor $\boldsymbol{\psi}_\Delta(\mathbf{y})$ to measure the score between $\mathbf{x}$ and $\mathbf{y}$ assuming a time-shift of Δ. We use this property to align the descriptors of $\mathcal{N}_1^{\mathrm{C}}$ in the temporal coordinate system of the query. In other terms, we modify the phase of the candidate i such that its similarity to the query is maximized with time-shift 0 instead of $\delta_{\mathbf{qy}}$. This is done by computing $\boldsymbol{\psi}_{-\delta_{\mathbf{qy}}}(\mathbf{y})$ from the stored vector $\boldsymbol{\psi}_0(\mathbf{y})$. The rationale is that we can now add these vectors to produce an augmented query that still offers localization capabilities. In contrast, we do not align the vectors of $\mathcal{N}_1^{\mathrm{U}}$, as they do not offer any temporal consistency.

4.2 Fusion strategy

We now consider several ways to produce an augmented query. They are inspired by other works on query expansion [6, 8]. The key differences are that we identify a time consistent subset with our triplet consistency and align these videos by performing a translation in the Fourier domain to take into account the relative time-shift with the query. We consider three fusion methods.

1. *Average Query Expansion (AQE).* The top $|\mathcal{N}_1|$ results associated with a query are aligned if time-consistent, and then averaged to produce the augmented query, as

$$\boldsymbol{\psi}^{\mathcal{C}}_{\mathrm{AQE}} = \frac{\boldsymbol{\psi}^{\mathcal{C}}(\mathbf{q}) + \sum_{\mathbf{x}\in\mathcal{N}_1} \boldsymbol{\psi}^{\mathcal{C}}(\mathbf{x})}{1+|\mathcal{N}_1|} \tag{24}$$

$$\boldsymbol{\psi}^{\mathcal{T}}_{\mathrm{AQE}} = \frac{\boldsymbol{\psi}^{\mathcal{T}}(\mathbf{q}) + \sum_{\mathbf{x}\in\mathcal{N}_1^{\mathrm{C}}} \boldsymbol{\psi}^{\mathcal{T}}_{-\delta_{\mathbf{qx}}}(\mathbf{x})}{1+|\mathcal{N}_1|} \tag{25}$$

where $\boldsymbol{\psi}^{\mathcal{C}}(\mathbf{x})$ is for the constant part of the kernel feature $\mathbf{x}$ and $\boldsymbol{\psi}^{\mathcal{T}}(\mathbf{x})$ is for the trigonometric part of the kernel feature $\mathbf{x}$. The final augmented query is the concatenation of these two features:

$$\boldsymbol{\psi}_{\mathrm{AQE}} = [\boldsymbol{\psi}^{\mathcal{C}}_{\mathrm{AQE}}, \boldsymbol{\psi}^{\mathcal{T}}_{\mathrm{AQE}}]. \tag{26}$$

2. *Difference of Neighborhood (DoN).* We combine our automatic alignment with an alternative QE method [8]. It is a variant of AQE where the average vector of close to far $|\mathcal{N}_2|$ neighbors is subtracted to the AQE vector.

$$\boldsymbol{\psi}_{\mathrm{DoN}} = \boldsymbol{\psi}_{\mathrm{AQE}} - \frac{\sum_{\mathbf{x}\in\mathcal{N}_2} \boldsymbol{\psi}_0(\mathbf{x})}{|\mathcal{N}_2|} \tag{27}$$

3. *DoN iterative.* The DoN is iteratively applied I times. Each time the new augmented query replaces the previous one. The score for one candidate for one iteration is averaged with the score at previous iteration.

5. DESIGN WITH MULTIPLE PERIODS

In this section, we propose an approach to improve the localization accuracy while keeping a reasonable size for the video representation. Our motivation is twofold:

- The localization can be done up to a period modulo T only. If a video is longer than this period, our method based on explicit feature maps can not disambiguate the relative offset.

- The localization accuracy depends on the period T (scale dependency). Increasing the modulation period dramatically reduces the localization accuracy for a given number of frequencies (*i.e.*, for a given size of the representation). In other terms, the localization is more precise for a shorter period.

We address these two problems by defining a temporal kernel constructed from multiple explicit feature maps, each being associated with a different period. As opposed to other contexts where the signal is inherently periodic, like angles [21], the choice of the period is a parameter of the method. We want to reduce it to achieve better localization. The main idea in this section is to use multiple elementary match kernels (with different periods), in order to still able to infer the correct time localization for long video.

About periodicity. The property motivating our proposal is that the sum of a T_1-periodic and of a T_2-periodic function is $T_1 \times T_2$-periodic if T_1 and T_2 are relatively prime. More generally, the period is $T_1 \times T_2/\gcd(T_1, T_2)$, where gcd is the greatest common divisor of T_1 and T_2. Said otherwise, if δ_t is a time shift such that $|\delta_t/2| \leq (T_1 \times T_2)/\gcd(T_1, T_2)$, we can determine δ_t if we know $\delta_t \mod T_1$ and $\delta_t \mod T_2$. This property can be generalized to more than two decompositions, which allows us to estimate time shifts for very long videos while keeping a small number of frequencies.

We apply this idea to the temporal match kernels introduced in the previous section to generate very long periods. Figure 4 shows a toy example with two temporal match kernels having distinct periods T_1 and T_2. We consider two synthetic vector sequences of 600 frames each having 100 similar frames in common: the subsequence $[100:200]$ in the first sequence corresponds to the subsequence $[400:500]$ in the second, which means a 300 frames temporal shift. We first present match kernels with periods $(T_1 = 100, T_2 = 200)$ to show that we obtain almost identical peaks due to resonance (top right figure). Unfortunately the higher peak happens at an offset of -500 frames. This is a pathological case that should be avoided because the induced period is only $T_1 \times T_2/\gcd(T_1, T_2) = 200$. In contrast, a proper choice of periods in the other example $(T_1 = 103, T_2 = 227)$ does not overlap and a single clear peak appear at 300 frames.

In practice, we use four periods of modulation, T_1 to T_4, all being prime numbers and therefore relatively prime to each other, such that $\gcd(T_1, T_2, T_3, T_4) = 1$. The combined kernel offers a long period. For each decomposition (each being associated with a different period), we compute the polynomial of score $\mathcal{K}_{\Delta,T}$ for each integer value on the interval $[-\max(L_1, L_2), +\max(L_1, L_2)]$, where L_1 and L_2 refer to the durations of the compared videos.

Two fusion operators are then considered to produce a global score function from the polynomial of scores associated with the temporal match kernel. The first is the sum of the polynomials, which itself can be written as a Fourier series. Other alternative are possible, such as taking minimum value of the different score functions. Experimentally,

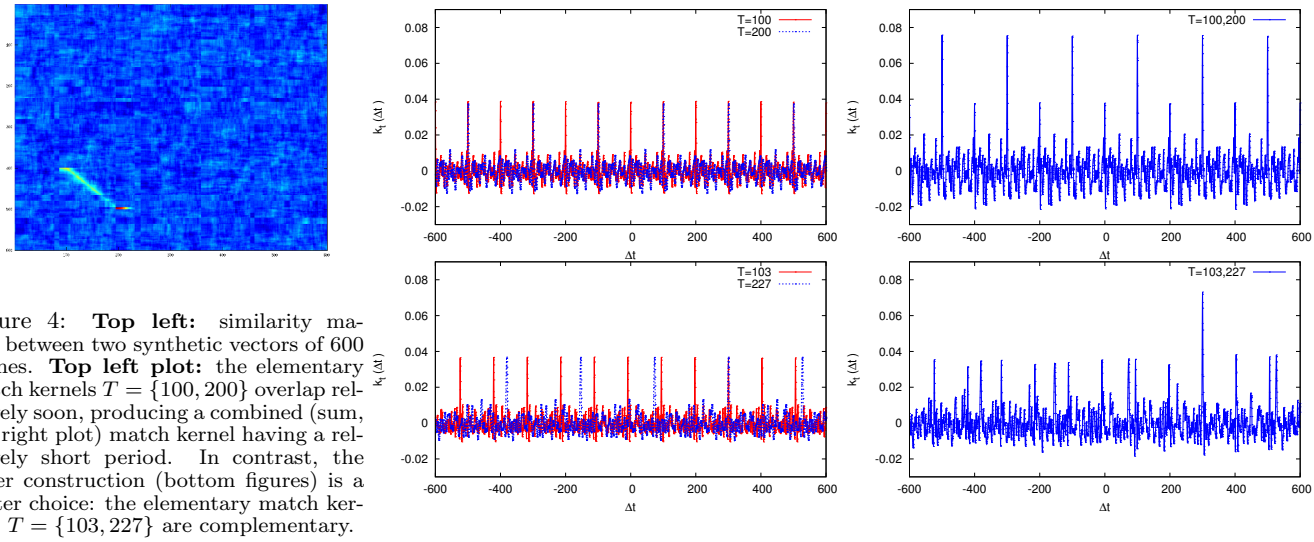

Figure 4: **Top left:** similarity matrix between two synthetic vectors of 600 frames. **Top left plot:** the elementary match kernels $T = \{100, 200\}$ overlap relatively soon, producing a combined (sum, top right plot) match kernel having a relatively short period. In contrast, the other construction (bottom figures) is a better choice: the elementary match kernels $T = \{103, 227\}$ are complementary.

the "sum" and "min" fusion perform similarly for video localization. Therefore we use the sum operator for this Multi-Period Strategy (MPS) in the experimental section.

Figure 9 shows the score function between two matching videos from the EVVE Madonna video set, using a single period $T = 65537$ or four periods $T = [2027, 3019, 5003, 7019]$ with the sum operator and two different settings $m = 4 \times 16$ *vs* $m = 1 \times 64$ and $m = 4 \times 32$ *vs* $m = 1 \times 128$. Observe that the multiple-periods strategy gives a better estimation of the offset. Although we do not have a precise temporal ground-truth on this dataset, we observe that this conclusion seems to be valid in most of the cases we checked: the multiple periods approach gives a better precision.

6. EXPERIMENTS

In this section, we evaluate the temporal match kernel introduced in Section 3, the multi-period variant of Section 5 and our temporally-aligned query expansion (Section 4) for recognition of particular events and for copy detection.

6.1 Event detection

Dataset. We consider the EVVE dataset [19] for particular event detection. The dataset contains videos from 13 particular events grabbed on Youtube™(see in Appendix 9 the details of the events). They are professional or private contents with strong or poor temporal consistency (details are given at the end of this subsection). 620 videos serve as queries, 2375 ones are forming the database.

Frame descriptors and baseline. We use the frame 1024-dimensional multi-VLAD whiten descriptors described in [19], available online, and compare with state-of-the-art methods that use the same input descriptors, namely CTE [19], Mean-MultiVLAD (MMV) [19] and Stable Hyper Pooling (SHP) [8]. CTE is able to provide localization, the two other ones can not. We also include the results achieved by MMV+CTE, which are obtained by adding the normalized scores from MMV and CTE for each query. Our method only involves three dependent parameters: β gives the steepness of the Gaussian approximated by the Fourier coefficient, T the period of modulation, and m the number of Fourier coefficients kept. This last parameter is the most important, as it induces the size of the representation. The method is not very sensitive to small variations of these values.

Event detection performance. Our method performs as well as CTE for the simple EVVE dataset, as shown in Table 1, while SHP remains higher but without alignment capability. Using a longer period of modulation, $T = 65537$ is preferable to a short one $T = 659$: a short period means that long videos will be "folded", consequently cumulated noise may create a peak in the polynomial score that overwhelms the true match. A longer period is safer but induces a less precise alignment for a fixed size of the description. When adding 100000 distractors to the database (+10K), our baseline is as good as MMV+CTE and very close to SHP. The checking of time consistency improves the precision of the method and makes it more robust in a noisy dataset.

Complexity. The modulation is done off-line for the collection. On average (over 10,000 videos, 30,000,000 frames), the modulation with $m = 32$ requires 0.3 s for a video clip comprising 3000 frames. Online, our Matlab implementation of the polynomial score computation requires 1 ms. For $m = 16$ the video descriptor dimensionality is 33792, twice bigger than SHP but still compact. The original CTE video representation depends on the video length [19], leading to a memory footprint of 943 MB for the whole collection, while our method is about 3 times more compact (320 MB).

Query expansion. We evaluate the query expansion methods detailed in Section 4. For this purpose, we consider the baseline descriptors with a unique modulation period $T = 65537$ and $m = 32$ frequencies. The experiments confirm that the proposed phase modification is compatible with the QE framework and does not degrade the results, while preserving the localization ability of the generated queries.

Table 2 compares the three query expansion methods proposed in section 4 to the baseline. Experimentally, parameters are set to $N_1 = 5$, $N_2 = 1000$, *AQE* and *DoN* require one iteration of expansion while *DoN iter* requires three iterations. AQE gives a good improvement but DoN strongly outperforms it. The benefit of the iterative approach is only 0.3% of mAP but is consistent over almost all events. Our best score after three iterations of query expansion is 41.3%. SHP [8] reported up to 44% on this dataset, CTE did not proposed any query expansion method, therefore no fair comparison can be provided (summary in Table 1).

In order to assess the scalability of the approach, 100000 distractors are added to the database (last line of table 1 and

Ours TE				
m	16		32	
T	659	65537	659	65537
EVVE	0.332	0.334	0.332	**0.335**

state of the art	CTE	MMV	CTE+ MMV	SHP	Ours TE
EVVE	0.352	0.334	**0.376**	0.363	0.335
+QE	-	-	-	**0.440**	0.413
+10K	0.202	0.220	0.254	**0.265**	0.254
+10K+QE	-	-	-	0.331	**0.347**

Table 1: **Top:** Results of our temporal embedding on the EVVE dataset, with two fixed periods of modulation $T = \{659, 65537\}$, two Fourier decompositions $m = \{16, 32\}$, with ($\beta = 32$). **Bottom:** state of the art methods VS ours ($m = 32$, $T = 65537$), with or without +10K distractors, with or without Query Expansion (simple *DoN* for SHP and phase modified *DoN* for Ours).

	baseline	AQE	DoN	DoN iter	+/-
#1	0.243	0.284	0.378	**0.393**	+ 0.15
#2	0.201	0.201	0.193	0.192	- 0.01
#3	0.088	0.056	0.104	0.104	+ 0.02
#4	0.120	0.116	0.151	0.156	+ 0.04
#5	0.235	0.241	0.277	**0.287**	+ 0.05
#6	0.340	0.406	**0.497**	**0.497**	+ 0.16
#7	0.140	0.043	0.110	0.111	- 0.03
#8	0.257	0.258	0.263	0.266	+ 0.01
#9	0.550	0.669	**0.762**	**0.764**	+ 0.21
#10	0.468	0.530	**0.588**	**0.581**	+ 0.11
#11	0.759	**0.815**	**0.814**	**0.806**	+ 0.05
#12	0.363	0.422	**0.484**	**0.494**	+ 0.13
#13	0.589	0.642	**0.712**	**0.721**	+ 0.13
mAP	0.335	0.361	0.410	0.413	+ 0.08

Table 2: Query expansion results – in bold the most significantly improved results for event detection. $N = 65537$, $m = 32$, $\beta = 32$, $N_1 = 5$, $N_2 = 1000$.

details in table 3). The mAP achieves 34.7% after 3 iterations, which is higher than the mAP=33.1% of SHP [8]. This suggests again that the temporal consistency is important to improve the accuracy of the search in larger collections.

Discussion. The events can be separated in two categories, depending on whether they are highly temporally consistent or not. The first group, namely events #$\{5, 6, 8, 9, 10, 11, 13\}$, mainly contains short live events (concerts, a royal weeding, a political speech, etc), while the second one (#$\{1, 3, 4, 7, 12\}$) essentially contains edited footage of news (riots, floods, bombing, etc). The improvement of mAP using the iterative query expansion (column *DoN iter*) is 9.5% (12.1% with the distractors) for the first group, 6.5% (7.6% with the distractors) for the second group, which clearly shows that our method takes advantage of the temporal consistency for generating more relevant expanded queries.

6.2 Localization by multi-period

We evaluate the quality of the localization on two benchmarks: 1) The video clip collection introduced for the copy detection task of the TRECVID 2011's evaluation campaign

	baseline	DoN	DoN iter	+/-
#1	0.219	**0.338**	**0.376**	+ 0.15
#2	0.178	0.173	0.184	+ 0.01
#3	0.077	0.076	0.084	+ 0.01
#4	0.070	0.090	0.093	+ 0.02
#5	0.183	0.204	0.219	+ 0.04
#6	0.248	**0.424**	**0.445**	+ 0.20
#7	0.109	0.091	0.131	+ 0.04
#8	0.228	0.234	0.250	+ 0.02
#9	0.458	**0.657**	**0.665**	+ 0.21
#10	0.451	**0.612**	**0.617**	+ 0.16
#11	0.331	0.373	**0.393**	+ 0.06
#12	0.206	**0.343**	**0.362**	+ 0.16
#13	0.541	**0.663**	**0.689**	+ 0.16
+10K mAP	0.254	0.329	0.347	+ 0.10

Table 3: Query expansion results with +10K – in bold the most significantly improved results for event detection. $N = 65537$, $m = 32$, $\beta = 32$, $N_1 = 5$, $N_2 = 1000$.

Figure 5: Examples of one video clip (reference is top-left) for which we find all its corresponding copies (5 transformations represented here). For this example, our technique finds the exact offsets (perfect temporal alignement).

[20] and 2) the Inria CLIMB dataset, which is specifically devoted to evaluate time synchronization in videos. The CTE [19] by Revaud *et al.* is used as the baseline. Recall that SHP [8] can not provide any localization.

TV CBCD 2011 dataset. This set was introduced for the copy detection task of the TRECVID 2011's evaluation campaign [20]. It contains 16776 reference videos and 1608 queries, all extracted at 30 frames per second. The queries were artificially created from 201 videos undergoing 8 different transformations. Some query videos have no corresponding match in the database, while for 134 groups of 8 queries the system is expected to identify the correct video clip and estimates the localization in time.

To evaluate the localization capability of our approach, we match each query with its original reference video. The estimated offset is compared to the ground-truth. We mention than the ground truth is not precise (up to one or two seconds of errors in some cases), likely because of some video encoding errors. Consequently we do not expect a precision at frame level. An example of typical transformations considered in the dataset are presented in Figure 5.

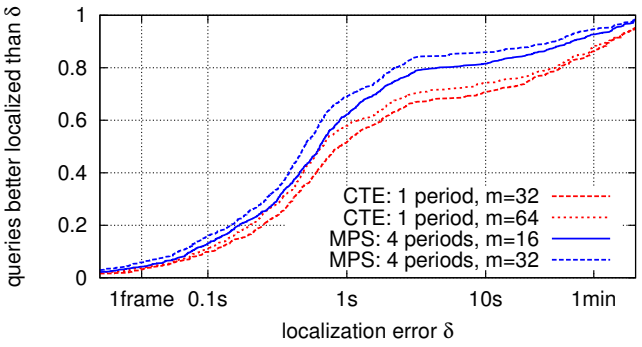

Figure 6: Localization accuracy on CBCD'2011 for CTE and our multi-period strategy (MPS). We show the rate of queries localized better than a tolerance δ (high is better). Note that the size of the representation for CTE with $m = 64$ is similar to that of MPS with 4 periods and $m = 16$.

Frame descriptors. We extract dense PCA-ROOT-sift vectors (step of 5 pixels) and compute an improved Fisher vector [17] for each frame (using a 128 Gaussian Mixture). The 8192-dimensional frame descriptors are further whitened and normalized [12] to produce vectors of 1024 components. As the longer database video clip contains 3694 frames, for the baseline we set $T = 3695$. For our multi-period approach we use the following periods: $\{1777, 2053, 2357, 2731\}$.

Localization performance. Figure 6 reports the localization performance for this copy detection task. Note that the frame rate is constant in this benchmark, which favors the temporal model assumed by CTE. Although for a few videos (about 15%), the localization accuracy is not good because the correct localization is only a local minimum of the score function and not the global one, overall, our multi-period approach clearly outperforms the single-period approach implemented in CTE, and this even when using a similar representation size. This is shown in the figure by comparing MPS with $m = 16$ frequencies against CTE with $m = 64$ frequencies. Using 4 times less frequencies compensates the fact that we use 4 periods w.r.t. complexity, yet gives much better localization accuracy. When manually looking at the alignment, we observe that our method often better aligns the videos than the ground-truth. The reader can also refer to figure 9, it shows how the multiple periods approach improves the localization for the EVVE dataset on a single real data example.

CLIMB dataset. This dataset has been created for the global video alignment task and is available online[2]. This dataset is also used for evaluating the precision of the alignment between pair of videos. The ground truth contains 89 videos at 30 fps. They have been shot, for each event, by different people moving around with different cameras. 452 temporal overlaps exist, however they are not necessarily visual overlaps. The ground truth is more precise than the TV CBCD 2011 one, here the location is at frame level.

The description is done as for the TV CBCD 2011 dataset. The videos of the dataset are longer than in the CBCD dataset, the longer one lasts 37771 frames. Consequently the single period search is set to $T = 37772$. The average video length being much longer than the CBCD dataset, for our MPS approach we set the periods to $\{2731, 4391, 9767, 14653\}$. The results for the localization are given in Figure 7 and

[2]http://pascal.inrialpes.fr/data/evve/index_align.html

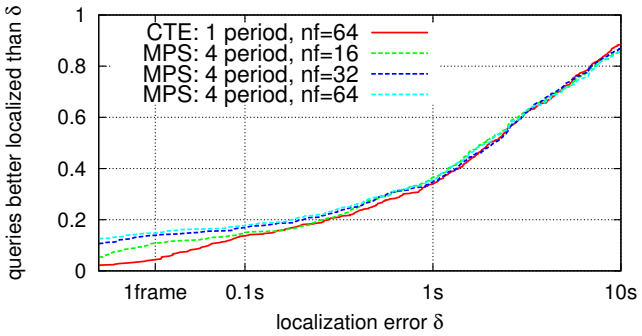

Figure 7: Localization accuracy on CLIMB dataset for CTE and our multi-period strategy (MPS). We show the rate of queries localized better than a tolerance δ (high is better). Note that the size of the representation for CTE with $m = 64$ is similar to that of MPS with 4 periods and $m = 16$.

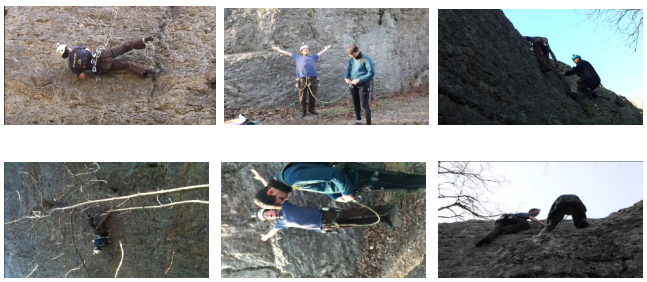

Figure 8: Captions of couples of video clips well aligned. One column, one couple. On the bottom, first and second ones were initially shot vertically.

illustrations of correctly aligned videos in Figure 8. The database is very challenging, yet the MPS approach clearly outperforms a single period approach and for a negligible extra computational cost (less than 10%). A direct comparison of CTE with $m = 64$ and our MPS method with $m = 4 \times 16$ ensures that the memory footprint is equivalent. In this case, our method is better, especially on small localization errors. Importantly, the accuracy can further be improved by giving more memory budget to our method, as shown when setting $m = 4 \times 32$ and $m = 4 \times 64$.

6.3 Query expansion with multiple periods

Finally, as a concluding experiment, we assess the complementarity of our methods on a event detection task when using all our contributions: the multiple periods match kernel combined with query expansion. The experiments are carried out on the EVVE dataset to allow a direct comparison. The results are reported on Table 4. Observe that they are almost identical to those reported for the single period match kernel with query expansion (Table 2). This suggests that the best option is indeed to combine these methods, as we obtain a similar detection rate with a better alignment.

7. CONCLUSION

This paper proposes a temporal matching kernel for efficient search and localisation of temporally consistent videos. The features, very compact and fast to compute, are also compatible with the query expansion strategies and yet preserve their localization capability. We further propose a multi-period strategy to improve the localisation of the common excerpts at not memory cost and for a very low extra

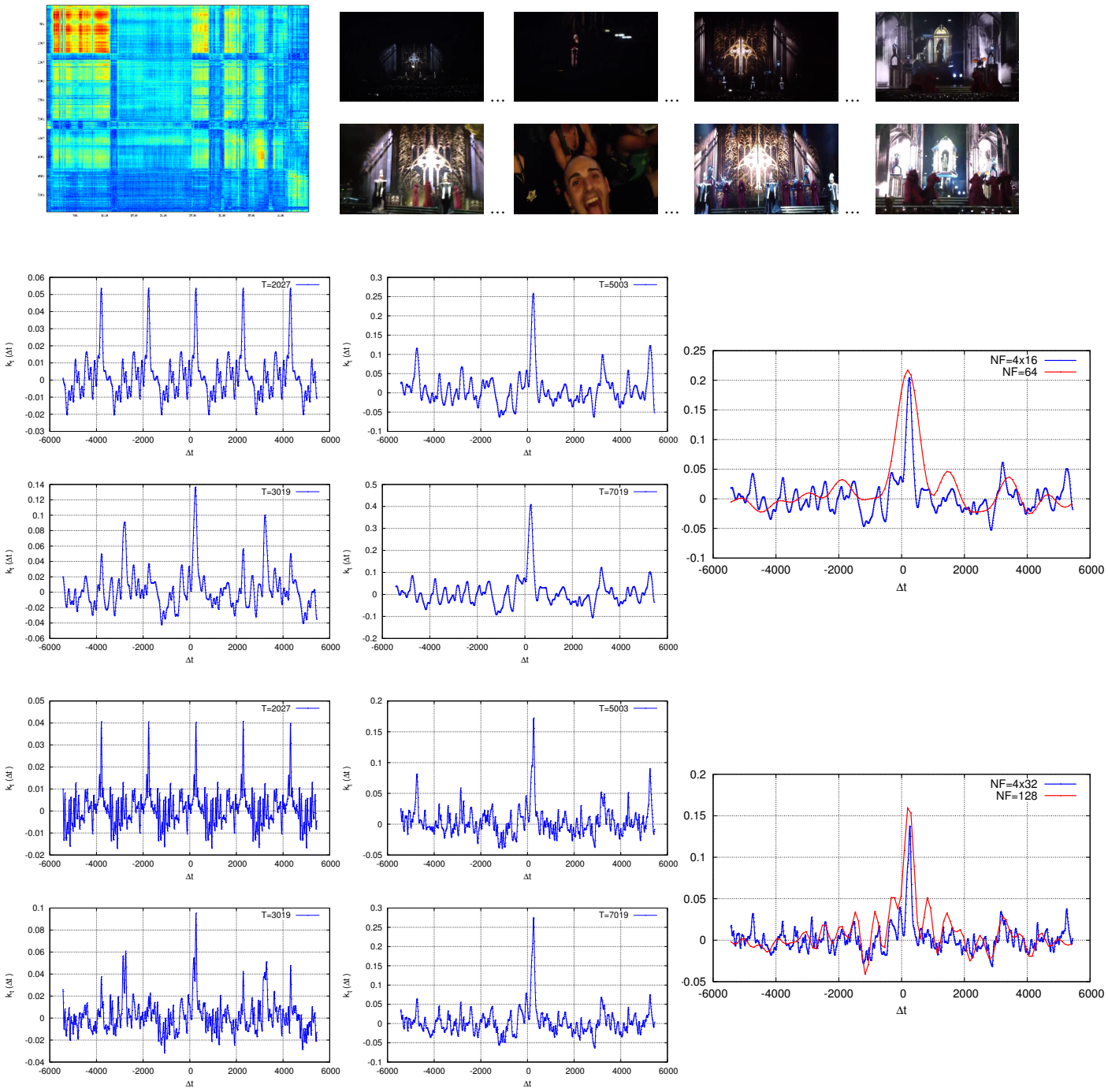

Figure 9: On the top row, on the first column the similarity matrix between two matching videos from the Madonna 2012 Roma concert event (4471 and 5441 frames), beside four captions of each video temporally aligned. It can be seen that it hard to evaluate a visual match between these videos: no clear ridge of high similarity appears in the matrix.
On the second row, the first four figures represent the result of the polynomial for some offsets between the videos for different periods of modulation $T = \{2027, 3019, 5003, 7019\}$ for $m = 16$. Their sum appear on the right figure in blue, while the result of the polynomial for a single period $T = \{65537\}$ for $m = 64$. The multi-period maximal value is observed at 16.3 seconds, the single period one at 12.8 seconds.
On the third row, are represented the same figures for $T = \{2027, 3019, 5003, 7019\}$ for $m = 32$. Their sum appear on the right figure in blue, while the result of the polynomial for a single period $T = \{65537\}$ for $m = 128$. The multi-period maximal value is observed at 17.1 seconds, the single period one at 12.8 seconds.
We visually estimate that the ground-truth offset is 17.5 seconds (+/- 0.5 second). The multi-period approach clearly improves the precision of the localisation.

	baseline	DoN	DoN iter	+/-
#1	0.242	0.383	**0.404**	+ 0.16
#2	0.200	0.194	0.193	- 0.01
#3	0.085	0.091	0.093	+ 0.01
#4	0.120	0.148	0.153	+ 0.03
#5	0.235	0.277	**0.286**	+ 0.05
#6	0.340	**0.502**	**0.498**	+ 0.16
#7	0.115	0.119	0.115	+ 0.00
#8	0.257	0.264	0.268	+ 0.01
#9	0.551	**0.762**	**0.764**	+ 0.21
#10	0.469	**0.587**	**0.580**	+ 0.11
#11	0.756	**0.812**	**0.803**	+ 0.05
#12	0.363	**0.484**	**0.494**	+ 0.13
#13	0.589	**0.714**	**0.723**	+ 0.13
mAP	0.332	0.410	0.413	+ 0.08

Table 4: Combining multi-period and query expansion on EVVE dataset – in bold the most significantly improved results for each event. $N = \{2027, 3019, 5003, 7019\}$, $m = 32$, $\beta = 32$, $N_1 = 5$, $N_2 = 1000$.

computation burden. All these items make the method effective for many tasks, as diverse as event retrieval, copy detection or video synchronization, and outperform state of the art methods that can compete on this range.

8. ACKNOWLEGMENT

This work was partly supported by ERC project Viamass no. 336054, JSPS KAKENHI Grant Numbers 26240016 and 24-02712, and NII International Internship.

9. ANNEX

The events are numbered as follows (alphabetical order): **1.** barcelona riots 2012 **2.** die toten hosen rock am ring 2012 **3.** dsk arrested **4.** egypt Tahrir Square protestors **5.** johnny stade de france 2012 **6.** kate william wedding **7.** marrakech bomb attack **8.** madonna rome 2012 **9.** obama speech victory **10.** shakira live kiev 2011 **11.** strokkur geyser **12.** thailand flood 2011 **13.** universal studios jurassic park ride

10. REFERENCES

[1] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, Jun. 2012.
[2] R. Arandjelovic and A. Zisserman. All about VLAD. In *CVPR*, Jun. 2013.
[3] L. Bo, X. Ren, and D. Fox. Kernel descriptors for visual recognition. In *NIPS*, Dec. 2010.
[4] L. Bo and C. Sminchisescu. Efficient match kernel between sets of features for visual recognition. In *NIPS*, Dec. 2009.
[5] O. Chum, A. Mikulik, M. Perdoch, and J. Matas. Total recall II: Query expansion revisited. In *CVPR*, Jun. 2011.
[6] O. Chum, J. Philbin, J. Sivic, M. Isard, and A. Zisserman. Total recall: Automatic query expansion with a generative feature model for object retrieval. In *ICCV*, Oct. 2007.
[7] M. Douze, H. Jégou, C. Schmid, and P. Pérez. Compact video description for copy detection with precise temporal alignment. In *ECCV*, Sep. 2010.
[8] M. Douze, J. Revaud, C. Schmid, and H. Jégou. Stable hyper-pooling and query expansion for event detection. In *ICCV*, Dec. 2013.
[9] M. K. et al. The visual object tracking VOT2014 challenge results. In *ICCV Workshops*, Jun. 2014.
[10] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *ECCV*, Oct. 2012.
[11] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista. High-speed tracking with kernelized correlation filters. *Trans. PAMI*, 2015. to appear.
[12] H. Jégou and O. Chum. Negative evidences and co-occurences in image retrieval: The benefit of PCA and whitening. In *ECCV*, Oct. 2012.
[13] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid. Aggregating local descriptors into compact codes. In *Trans. PAMI*, Sep. 2012.
[14] H. Jégou and A. Zisserman. Triangulation embedding and democratic kernels for image search. In *CVPR*, Jun. 2014.
[15] J. Law-To, L. Chen, A. Joly, I. Laptev, O. Buisson, V. Gouet-Brunet, N. Boujemaa, and F. Stentiford. Video copy detection: a comparative study. In *CIVR*, pages 371–378, 2007.
[16] F. Perronnin and C. R. Dance. Fisher kernels on visual vocabularies for image categorization. In *CVPR*, Jun. 2007.
[17] F. Perronnin, J. Sánchez, and T. Mensink. Improving the Fisher kernel for large-scale image classification. In *ECCV*, Sep. 2010.
[18] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman. Lost in quantization: Improving particular object retrieval in large scale image databases. In *CVPR*, Jun. 2008.
[19] J. Revaud, M. Douze, C. Schmid, and H. Jégou. Event retrieval in large video collections with circulant temporal encoding. In *CVPR*, Jun. 2013.
[20] A. F. Smeaton, P. Over, and W. Kraaij. Evaluation campaigns and trecvid. In *MIR*, pages 321–330, 2006.
[21] G. Tolias, T. Furon, and H. Jégou. Orientation covariant aggregation of local descriptors with embeddings. In *ECCV*, Sep. 2014.
[22] G. Tolias and H. Jégou. Visual query expansion with or without geometry: refining local descriptors by feature aggregation. *Pattern Recognition*, Apr. 2014.
[23] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *Trans. PAMI*, 34(3):480–492, Mar. 2012.
[24] J. Wang, J. Yang, F. L. K. Yu, T. Huang, and Y. Gong. Locality-constrained linear coding for image classification. In *CVPR*, Jun. 2010.
[25] M.-C. Yeh and K.-T. Cheng. Video copy detection by fast sequence matching. In *CIVR*, 2009.