Efficient motion estimation methods for fast recognition of activities of daily living

Stergios Poularakis, Konstantinos Avgerinakis, Alexia Briassouli*, Ioannis Kompatsiaris

CERTH – ITI, 6th km Charilaou-Thermi rd Thermi, Thessaloniki, Greece

ARTICLE INFO

Keywords:
Activity recognition
Motion estimation
Block matching
MPEG video encoding
Computational efficiency

ABSTRACT

This work proposes a framework for the efficient recognition of activities of daily living (ADLs), captured by static color cameras, applicable in real world scenarios. Our method reduces the computational cost of ADL recognition in both compressed and uncompressed domains by introducing system level improvements in State-of-the-Art activity recognition methods. Faster motion estimation methods are employed to replace costly dense optical flow (OF) based motion estimation, through the use of fast block matching methods, as well as motion vectors, drawn directly from the compressed video domain (MPEG vectors). This results in increased computational efficiency, with minimal loss in terms of recognition accuracy. To prove the effectiveness of our approach, we provide an extensive, in-depth investigation of the trade-offs between computational cost, compression efficiency and recognition accuracy, tested on bench-mark and real-world ADL video datasets.

1. Introduction

The recognition of Activities of Daily Living (ADLs) has drawn significant research attention in the computer vision community, as their monitoring can provide valuable information for applications such as assisted living, remote healthcare, lifestyle and behavioral profiling. To this end, efforts are being made to design algorithms of ADL recognition which are both accurate and computationally efficient. While a plethora of activity recognition methods exist, most overlook the importance of computational and compression efficiency, focusing only on recognition accuracy. The main bottleneck of current State-of-the-Art (SoA) works [1–7] is the use of computationally expensive Optical Flow (OF) [8] for motion estimation and feature extraction.

This work addresses the issue of computational efficiency by expanding upon our previous work in [9]: computationally costly dense OF is replaced by computationally lighter Block Matching (BM) and MPEG encoded motion vectors for activity recognition. The motion field is post-processed and its results are incorporated in a dense, trajectory-based activity recognition framework [1]. In-depth experiments on benchmark ADL video datasets compare some of the most reliable and popular OF and BM methods, as well as the most common encoded motion vectors, demonstrating that the latter increase computational efficiency at a minimal loss in recognition accuracy, compared to related work.

Recent works [10–13] also used motion vectors drawn directly from the MPEG compressed video domain, resulting in a significant computational speedup (~66%) at a small reduction of recognition accuracy (~5%). We extend these works by providing a thorough comparison of very popular video compression standards, applied specifically for the recognition of ADLs, in contrast to more generic works [14–16]. We also investigate various configuration parameters of MPEG video encoding, such as GOP size and the motion estimation algorithm used to examine their effect on recognition accuracy and compression efficiency (video quality and bit rate) and identify those that make a difference in the measured performance. We finally use the precomputed MPEG motion vectors to seed and accelerate the BM search. This analysis reveals trade-offs between bit rate (file size), PSNR (video quality), ADL recognition accuracy, resulting in useful guidelines to practitioners. In short, our contributions are:

- A framework for efficient recognition and coding of human activities in video by exploring the trade-off s at all stages: (1) video compression efficiency, (2) computational efficiency, (3) recognition accuracy.
- We propose and evaluate the use of existing compressed motion vectors in conjunction with BM, for improved, faster recognition accuracy and computational savings.
- A thorough Rate-Distortion-based comparison (bitrate-video quality) between very popular video compression standards, applied specifically to activities of daily living.

The rest of this paper is organized as follows: In Section 2 we review

*Corresponding author.
E-mail addresses: stpoular@iti.gr (S. Poularakis), koafgeri@iti.gr (K. Avgerinakis), abria@iti.gr (A. Briassouli), ikom@iti.gr (I. Kompatsiaris).
http://dx.doi.org/10.1016/j.image.2017.01.005
Received 20 January 2016; Received in revised form 8 November 2016; Accepted 18 January 2017
Available online 26 January 2017
0923-5965/ © 2017 Elsevier B.V. All rights reserved.
related SoA approaches for activity recognition and motion estimation. In Section 3 we present our activity recognition framework method in detail, including the motion estimation methods used. Several aspects of video encoding examined are detailed in Section 4, namely the effectiveness of different video codecs, and various encoding parameters. Experimental results are presented in detail in Section 5, comparing OF with BM, BM with BM seeded by MPEG vectors, the computational efficiency of all methods, and a joint performance metric. Finally, Section 6 concludes this paper and addresses our plans for future work.

2. Related work

2.1. Activity recognition methods

Numerous approaches have been developed in recent years for activity recognition. Our work is closely related to methods based on trajectories of interest points [1,3–7,17–21], which can be roughly divided into those where interest points are sampled sparsely or densely. The approaches of the first category extract sparse interest points via standard interest point detectors and track them over time. Messing et al. [17] tracked corner points using the KLT tracker [22] for trajectories of log-polar quantized velocities. They then trained directed graphical models using these trajectories for ADL recognition. In a similar approach, Sun et al. [18] used trajectories of tracked SIFT points [23] with stationary Markov chains, while Matikainen et al. [19] performed trajectory clustering in a bag-of-words framework.

In the second category, densely sampled points are tracked using OF [8] and global smoothness constraints. In [24], dense features are detected, tracked and encoded via motion prediction. Dense features are then used for efficient and ac-curate object recognition. In their pioneering work on human action recognition, Wang et al. [1] sampled feature points on a grid and tracked them over multiple spatial scales using dense OF. Subsequently, they computed HOGHOF descriptors [2] in a fixed space-time volume around each trajectory, and described the OF using the Motion Boundary Histogram (MBH) [25]. The use of dense trajectories led to SoA recognition accuracy, however the computation of dense OF consumed over 50% of the total run-time [3]. In an improved version, Wang et al. [4] improved tracking robustness through SURF point [26] matching, resulting in increased recognition accuracy, but at an even higher computational cost. Jain et al. [5] further improved recognition accuracy through VLAD trajectory encoding [27], while they also used motion compensation with a homographic model to overcome camera motion and eliminate background trajectories. Oneata et al. [21] encoded trajectories of dense SIFT points using a Fisher encoding scheme [28], and included weak spatio-temporal location information in their representation. Jiang et al. [6] used global reference points to encode single trajectories and local reference points to describe relations between pairs of trajectories. Ochs et al. [7] used large displacement OF [29] to form well defined trajectory structures, which were tracked over videos of a long duration, avoiding track drifting. Recently, activity of daily living recognition algorithms have also been developed for the challenging problem of activity recognition from wearable cameras in [30] and, more recently via a weakly supervised technique that uses multilask clustering to group similar activities that a person performs during the day to extract behavioral patterns [31]. Notable activity recognition techniques have been also introduced in [32], where human pose is recovered from video frames using non-linear mapping with multi-layered deep neural networks, and in [33], where sparse coding and a local similarity preserving term are used to recover 3-D human poses from silhouettes. Further advances in human pose recovery have been also noted in [34], where back-propagation deep learning is used in order to learn a non-linear mapping from 2D images to 3D poses.

2.2. Efficient motion estimation

Motion estimation is the main bottleneck of most activity recognition approaches, taking up about 50–60% of the computational time [3,13], due to the high cost of dense OF estimation [22]. Some efforts have focused towards increasing computational efficiency by: 1) using newer, more computationally efficient OF methods [35–39] or 2) retrieving the motion vectors from the compressed video domain [10–13].

Among the newer and faster methods for OF estimation, local and global concepts are combined in a single scheme in [36], based on the differential estimation of OF vectors and the spatial smoothing of the vectors in neighboring regions. This produces a dense OF matrix for fast motion vector estimation. Farneback et al. [8] first computed 3D orientation tensors from the image sequence and then applied parametric motion model constraints for accurate velocity estimates in a computationally efficient manner. Accurate and dense OF matrices are estimated in [37] using Total Variation regularization and the L1 norm in the data fidelity term (TV-L1) for increased robustness to illumination changes, occlusions and noise at a high computational cost. Recently, Sundaram et al. [38] accelerated OF methods through GPUs, while Tao et al. [39] proposed a locally computed probabilistic representation of motion with running times in-creasing sub-linearly with the number of pixels. However these methods rely on hardware solutions for faster motion estimation, rather than algorithmic ones, as we explain in detail below.

The second category of methods leverages pre-computed motion vectors from the compressed video domain, which are typically produced through block matching applied to uncompressed video data, and are then stored in the com-pressed video. Depending on the type of visual features (motion, appearance) used for activity representation, full video decoding may not be necessary, thus saving a significant amount of computational time [13].

In an early work, Babu and Ramakrishnan [10] used the MPEG motion vectors to compute Motion History Image [40] and Motion Flow History features with good activity recognition results. Yeo et al. [11] performed action recognition using the H.264 motion vectors, investing the effect of various compression parameters (size and structure of GOP, block size and motion estimation accuracy) on the recognition accuracy. However, they did not provide an in-depth, thorough comparison with existing OF-based methods. Recently, Biswas and Babu [12] represented videos as space-time cubes and computed Histograms of Oriented Motion Vectors, using the motion vectors of H.264 compressed videos. Subsequently, they performed video classification under a typical Bag-of-Words framework, achieving near state-of-the-art results.

Kantorov and Laptev [13] computed HOF and MBH features using the MPEG motion vectors and performed activity recognition using Fisher vector coding and SVMs. Their main result was that compressed motion vectors offer a significant computational speedup (~66%) with a small reduction in recognition accuracy (~5%), while additional optimizations of the encoding (VLAD [27]) and classification (Approximate Nearest Neighbor [41]) methods used further increase computational efficiency. With the exception of [11], these works did not take into account video encoding parameters, nor their effect on recognition accuracy and compression efficiency.

GPU programming is increasingly used today, as it can improve the computational cost of challenging algorithms, which require the extraction of large amounts of feature vectors as in image processing. We tested GPU programming to speed up the estimation of motion vectors, however this proved to be computationally expensive as well, cancelling out the improvements brought by the faster processing speed. This is because GPUs require the transfer of large amounts of data in the GPU memory, since they do not analyze and encode the video directly in the RAM.

It should also be noted that, despite their effectiveness, GPU-based
دریافت فوری متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارتهای عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات