Dynamic Scene Recognition with Complementary Spatiotemporal Features

Christoph Feichtenhofer Axel Pinz Richard P. Wildes

Abstract—This paper presents Dynamically Pooled Complementary Features, a unified approach to dynamic scene recognition that analyzes a short video clip in terms of its spatial, temporal and color properties. The complementarity of these properties is preserved through all main steps of processing, including primitive feature extraction, coding and pooling. In the feature extraction step, spatial orientations capture static appearance, spatiotemporal oriented energies capture image dynamics and color statistics capture chromatic information. Subsequently, primitive features are encoded into a mid-level representation that has been learned for the task of dynamic scene recognition. Finally, a novel dynamic spacetime pyramid is introduced. This dynamic pooling approach can handle both global as well as local motion by adapting to the temporal structure, as guided by pooling energies. The resulting system provides online recognition of dynamic scenes that is thoroughly evaluated on the two current benchmark datasets and yields best results to date on both datasets. In-depth analysis reveals the benefits of explicitly modeling feature complementarity in combination with the dynamic spacetime pyramid, indicating that this unified approach should be well-suited to many areas of video analysis.

Index Terms—Dynamic scenes, feature representations, visual spacetime, image dynamics, spatiotemporal orientation

1 Introduction

Video analysis is a highly researched area and currently there is an enormous interest in spacetime analysis at various levels of complexity, ranging from optical flow and dynamic texture analysis to high-level analysis in terms of actions, activities and localization of particular events in videos. While the target application in this paper is dynamic scene recognition, at the same time this paper contributes a principled, well-founded suite of representations and algorithms with potential to benefit spacetime analysis at all levels of abstraction.

Dynamic scenes are characterized by a collection of dynamic patterns and their spatial layout, as captured in short video clips. For instance, a beach scene might be characterized by drifting overhead clouds, mid-scene water waves and a foreground of static sandy texture. Other examples include forest fires, avalanches and traffic scenes. These scenes may be captured by either stationary or moving cameras; thus, while scene motion is characteristic, it is not exclusive of camera induced motion. Indeed, dynamic scene classification in the presence of camera motion has proven to be more challenging than when this confounding attribute is absent. In comparison, dynamic textures (*e.g.*, [1, 2, 3]) also are concerned with complicated dynamic patterns, but in simpler settings, typically with stationary cameras and the field of view completely occupied by the particular complex dynamic pattern.

Currently, there exist two benchmark datasets of dynamic scene videos with [4], and without camera motion [5]; the proposed approach is evaluated on both of these datasets. Figure 1 shows typical images from a few categories of these datasets, illustrating the challenges of small inter-class differences (e.g., a waterfall may appear very similar to a fountain) as well as large intra-class variations. Interestingly, humans are able to perform dynamic scene recognition quickly and accurately [6, 7], and with little attention paid to the objects present in the scene [8], which also makes automated dynamic scene recognition an attractive goal in itself. Furthermore, dynamic scene recognition can provide relevant contextual information in many other applications of video analysis (*e.g.*, scene context can improve human action recognition [9]).

In this paper, dynamic scene recognition is tackled in accord with the dominant approach to image classification and object recognition via three steps: feature extraction, coding and pooling. As pointed out in [10] for the case of the coding step, "the devil is in the details", and careful choice of methods and parameters is crucial for success of all three steps. Figure 2 provides an overview of the present approach, which is termed "Dynamically Pooled Complementary Features" (DPCF). An input video is analyzed in *slices* of the sequence, which are defined as short temporal intervals. For each slice, at each spatiotemporal location $\mathbf{x} = (x, y, t)^\top$, complementary spatial $\mathbf{v}_\mathrm{S}$, temporal $\mathbf{v}_\mathrm{T}$, and color $\mathbf{v}_\mathrm{C}$ features are extracted. Next, these features are encoded into a mid-

• *C. Feichtenhofer and A. Pinz are with the Institute of Electrical Measurement and Measurement Signal Processing, TU Graz, Austria. R. Wildes is with the Department of Electrical Engineering and Computer Science and the Centre for Vision Research, York University, Toronto, Canada.*
E-mail: {feichtenhofer, axel.pinz}@tugraz.at wildes@cse.yorku.ca.

(a) Images from three different classes with similar appearance.

(b) Images from landslide sequences with large differences in appearance.

Fig. 1: Examples for small inter class differences (a) and large intra class variations (b) from the YUPENN [5] (a) and the Maryland [4] (b) datasets.

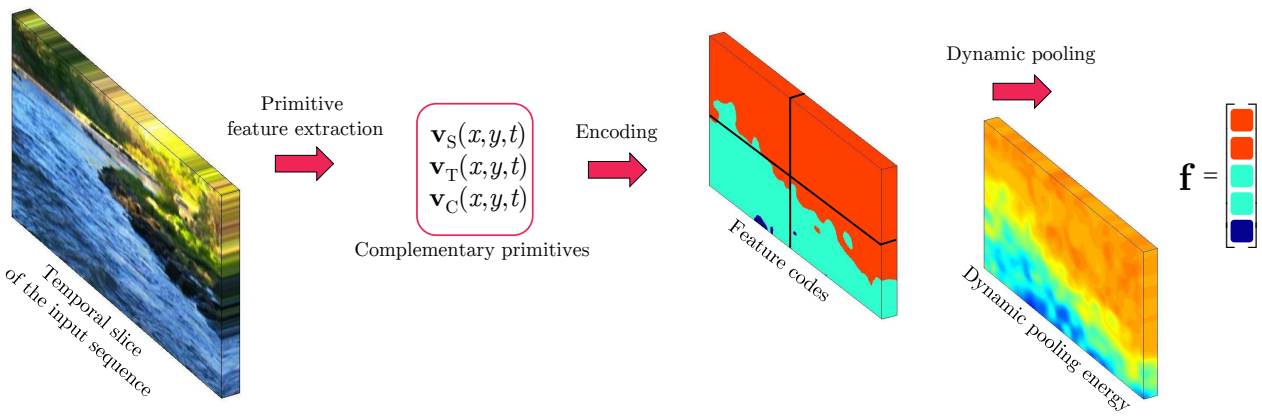


Fig. 2: Dynamically Pooled Complementary Features (DPCF) Overview. First, spatial $\mathbf{v}_S$, temporal $\mathbf{v}_T$, and color $\mathbf{v}_C$ features are extracted at each spatiotemporal location $\mathbf{x} = (x, y, t)^\top$ from a temporal slice of the input video. Second, features are encoded into a mid-level representation learned for the task and also used to extract dynamic pooling energies. Third, the encoded features are pooled via a novel dynamic spacetime pyramid that adapts to the temporal image structure, as guided by the pooling energies. The pooled encodings are concatenated into vectors that serve as the final representation for online recognition.

level representation that has been tuned to the task of dynamic scene recognition via a training procedure. Finally, the encoded features are pooled by a novel, dynamic spacetime pyramid that adapts to temporal scene dynamics, resulting in a feature vector, $\mathbf{f}$, that is subject to online classification. Complementarity in terms of spatial, temporal and color channels is preserved through all three steps of processing. Note that the approach is also strongly supported by findings from neurobiology of natural visual systems [11], where there is a separation into parvocellular, magnocellular and konio layers that strongly suggests a similar complementarity (spatial, motion and color channels) of visual pathways.

The DPCF approach builds in previous research by the authors [23, 24], yet provides significant advances as detailed in Section 3. In empirical evaluation (see Section 4), DPCF greatly outperforms both these previous approaches to dynamic scene recognition to set a new state-of-the-art.

2 Related Work

Significant research has considered scene recognition from single images (*e.g.*, [12, 13, 14, 15, 16, 17, 18, 19, 20, 21]). In contrast, relatively little attention has been paid to dynamic scene recognition from temporal sequences of images (e.g., video), even though temporal information should support additional representational richness for visual scene classification. A possible reason for this limited research in dynamic scenes was the lack of a substantial database; however, this limitation has now been addressed, as two dynamic scene video databases have appeared

[4, 5]. Correspondingly, a literature on dynamic scene recognition is emerging.

Video-based dynamic scene classification was first introduced in the context of human action recognition, where it was shown that automatically extracted scene context information can improve action recognition [9]. Histograms of optical flow were used to characterize both human actions and scene dynamics. Based on success in modeling dynamic textures with linear dynamical systems [2, 22], other research has considered such measurements for dynamic scene recognition [4]. The same paper also presented an alternative approach that fuses static and dynamic features in a chaos theoretic approach. Empirical evaluation showed that the chaos theoretic approach outperformed the dynamical systems approach in classification of "in-the-wild" dynamic scenes. These relative performance results may be accounted for by the limitations of the first-order Markov property and linearity assumptions inherent in the dynamical systems approach.

By analogy with the important role that purely spatial orientation primitives can play in static scene recognition (*e.g.* [12, 13, 16]), research in dynamic scene recognition has generalized to exploiting measures of spatiotemporal orientation [5, 23, 24]. Other work has investigated slow feature analysis (SFA) [25] applied to purely spatial orientation measurements [26] for dynamic scene recognition [27]. More generally, spacetime oriented energies have shown their virtues in many areas of video analysis, including dynamic texture recognition [3], scene recognition [5], target tracking [28], human action recognition [29], anomaly detection [30] and spacetime stereo [31].

A useful perspective on these various research strands is provided by a study [5] that systematically investigated the impact of primitive feature representations as well as the utility of spatial pyramids [16] for dynamic scene recognition. The study compared spatial appearance, temporal dynamics and joint appearance/dynamics features to conclude that features that jointly model spatial appearance and temporal dynamics in conjunction with spatial pyramids provided overall best performance for recognizing dynamic scenes.

3 Dynamically Pooled Complementary Features – DPCF

This section details the Dynamically Pooled Complementary Features (DPCF) approach to spacetime image representation in application to dynamic scene recognition. The description unfolds in accord with the tripartite paradigm of primitive feature extraction, encoding and pooling.

The approach leverages two ideas from previous work by the authors in application to dynamic scene recognition: Use of complementary features [23]; operation in a bag of visual words framework using a dynamc pyramid for pooling [24]. Otherwise, the approach is significantly different from its predecessors, as follows. First, the primitive features used are different. The previous work using complementary features employed different spatial, temporal and chromatic features [23]. The other work [24] did not use complementary spatial and temporal features at all. Second, DPCF aggregates its primitve feature descriptors in spatiotemporal grids to capture neighborhood structure. Neither of the previous approaches explicitly modeled neighborhood structure. Third, DPCF encodes features in terms of Fisher vectors. In contrast, the previous approaches either did not peform explicit encoding at all [23] or did so via LLC encoding [24]. Finally, in classification DPCF combines its complementary cues via late fusion of their respective support vector machine (SVM) scores, while [24] used a single SVM to classify its feature vector and [23] used a random forest classifier.

3.1 Primitive feature extraction

The developed descriptor for dynamic scene representation is based on the complementary combination of several different primitive measurements. Spatially oriented measurements are used to capture static image appearance and are combined with temporal oriented measurements to capture image dynamics. Additionally, color channels are included to capture complementary chromatic information. Thus, the features are complementary by construction. Interestingly, evidence from biological systems suggests that they exploit similar complementary feature combination in their visual processing [32, 33, 34].

3.1.1 Spatial features

Spatial appearance information is extracted via application of multiscale derivative filters that are tuned for spatial orientation. Here, 2D Gaussian first derivative filters, $G_{2D}^{(1)}(\theta_i, \sigma_j) = \kappa \frac{\partial}{\partial \theta_i} \exp\left(-\frac{x^2+y^2}{2\sigma_j^2}\right) = -\kappa(x\cos\theta_i + y\sin\theta_i)\exp\left(-\frac{x^2+y^2}{2\sigma_j^2}\right)$, with κ a normalization factor, θ_i denoting orientation and σ_j scale, are applied to yield a set of multiscale, multiorientation measurements according to

$$E_{\mathrm{S}}(\mathbf{x}; \theta_i, \sigma_j) = \sum_{\Omega} G_{2D}^{(1)}(\theta_i, \sigma_j) * \mathcal{I}(\mathbf{x}), \qquad (1)$$

where $\mathcal{I}$ is an image, $\mathbf{x} = (x, y)^\top$ spatial coordinates, $*$ convolution, Ω a local aggregation sub-region and subscript S appears on E_{S} to denote *spatial* orientation.

Figure 3 shows the spatial filtering results on a windmill farm sequence. Notice, *e.g.*, how the vertical structure of the windmill bases yields largest magnitude responses for the correspondingly oriented filter,

$\theta_3 = 90°$, while the different orientations of the windmill blades are preferentially enhanced according to the most closely matched filter orientations.

3.1.2 Temporal features

Similarly, dynamic information is extracted via application of 3D Gaussian third derivative filters, $G_{3D}^{(3)}(\theta_i, \sigma_j) = \kappa \frac{\partial^3}{\partial \theta_i^3} \exp\left(-\frac{x^2+y^2+t^2}{2\sigma_j^2}\right)$, with θ_i now denoting the 3D filter orientation (*e.g.*, given in terms of direction cosines and $G_{3D}^{(3)}$ expanded analgously to $G_{2D}^{(1)}$ above) and σ_j scale,

$$E_{\mathrm{ST}}(\mathbf{x}; \theta_i, \sigma_j) = \sum_{\Omega} |G_{3D}^{(3)}(\theta_i, \sigma_j) * \mathcal{V}(\mathbf{x})|^2, \qquad (2)$$

with the grayscale spacetime volume, $\mathcal{V}$, indexed by $\mathbf{x} = (x, y, t)^\top$, formed by stacking all video frames of a sequence along the temporal axis, t, and Ω being the aggregation sub-region. Subscript ST on E_{ST} denotes *spatiotemporal* orientation.

Following previous work in spacetime texture analysis [3], the spatiotemporal responses, (2), are further combined to yield measures of dynamic information independent of spatial appearance, as follows. In the frequency domain, motion occurs as a plane through the origin [35, 36]. In particular, an image suffering a uniform translation, $\mathbf{v} = (u, v, 1)^\top$, can be written as $I(\mathbf{x}) = I(x - ut, y - vt, t)$ and its corresponding spectrum is given as $\tilde{I}(\mathbf{k}) = \tilde{I}(\omega_x, \omega_y)\delta(\omega_x u + \omega_y v + \omega_t)$, where $\mathbf{k} = (\omega_x, \omega_y, \omega_t)^\top$ denotes the spatiotemporal frequency vector, $\tilde{I}$ denotes the Fourier transform of I and $\delta(\cdot)$ is the Dirac delta function. Geometrically, $\tilde{I}(\mathbf{k})$ can be interpreted as the spectrum being restricted to a plane through the origin with normal $\mathbf{v}$. Correspondingly, summation across a set of $x - y - t$ energy measurements consistent with a single frequency domain plane through the origin is indicative of the associated spacetime orientation, independent of purely spatial orientation.

Let the plane, $\Pi(\hat{\mathbf{n}})$, be defined by its normal, $\hat{\mathbf{n}} = (n_x, n_y, n_t)^\top$, then measurements of orientation consistent with this plane are given as

$$E_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_k, \sigma_j) = \sum_{\theta_i \in \Pi(\hat{\mathbf{n}}_k)} E_{\mathrm{ST}}(\mathbf{x}; \theta_i, \sigma_j), \qquad (3)$$

with θ_i one of $N + 1$ equally spaced orientations consistent with the frequency domain plane Π and $N = 3$ is the order of the employed Gaussian derivative filters; for details see [3]. Here, the subscript T on E_{T} serves to denote that the spatiotemporal measurements have been "*marginalized*" with respect to purely spatial orientation.

As noted in Sec. 1, previous spacetime filtering approaches to dynamic scene recognition tend to exhibit decreased performance when dealing with scenes captured with camera motion, in comparison to scenes captured with stationary cameras. A likely explanation for this result is that the approaches have difficulty in disentangling image dynamics that are due to camera motion vs. those that are intrinsic to the scenes. Here, it is interesting to note that camera motion often unfolds at coarser time scales (*e.g.*, extended pans and zooms) in comparison to intrinsic scene dynamics (*e.g.*, spacetime textures of water, vegetation, *etc.*); however, previous approaches have made their measurements using relatively coarse temporal scales and thereby failed to exploit this difference. In the present approach this difference in time scale is captured by making use of only fine scales, σ, during *temporal* filtering, (2), so that they are preferentially matched to scene, as opposed to camera, dynamics.

Owing to the bandpass nature of the Gaussian derivative filters, the orientation measurements are invariant to additive photometric variations (*e.g.*, as might arise from local image brightness change in imaged scenes). To provide additional invariance to multiplicative photometric variations for the dynamic measurements, (3), each motion direction selective measurement is normalized with respect to the sum of all filter responses at that point according to

$$\hat{E}_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_k, \sigma_j) = \frac{E_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_k, \sigma_j)}{\sum_{l=1}^{M} E_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_l, \sigma_j) + \epsilon}, \qquad (4)$$

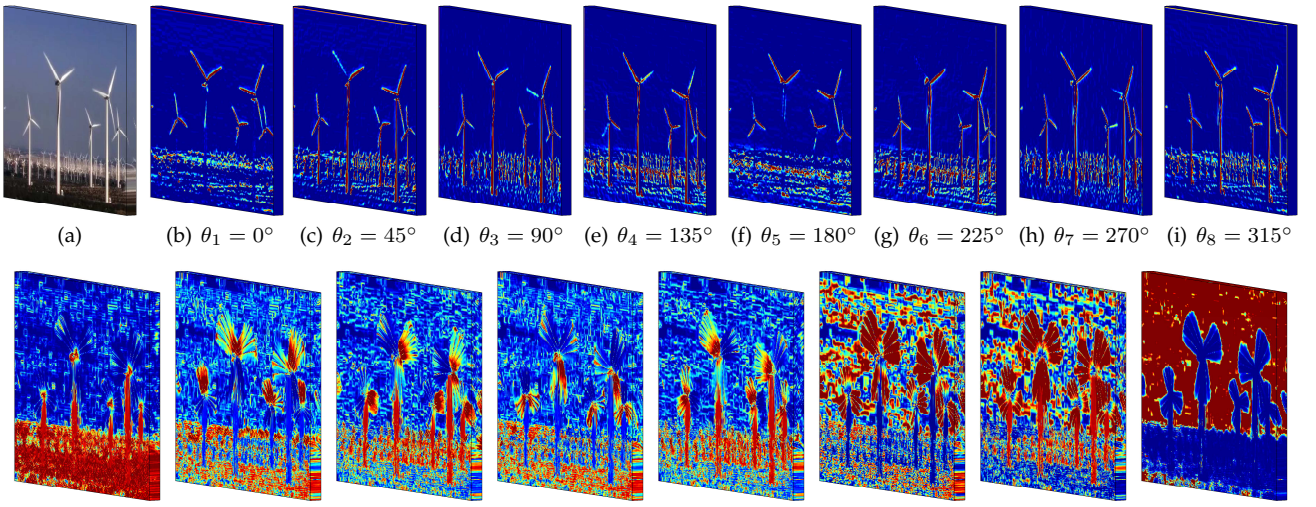
where M denotes the number of temporal orientation measurements considered to yield a normalized set of measurements, $\hat{E}_{\mathrm{T}}$. Note that ϵ is a small constant added to the sum of the energies over all orientations. This bias operates as a noise floor and avoids numerical instabilities at low overall energies. (See, *e.g.*, [37] for more general discussion of photometrically invariant features.) To explicitly capture lack of oriented spacetime structure, another feature channel

$$\hat{E}_{\mathrm{T}}^{\epsilon}(\mathbf{x}; \sigma_j) = \frac{\epsilon}{\sum_{l=1}^{M} E_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_l, \sigma_j) + \epsilon}, \qquad (5)$$

is added to the contrast-normalized filter responses of (4). Note, *e.g.*, that regions lacking oriented structure will have the summation in (5) evaluate to 0; hence, $\hat{E}_{\mathrm{T}}^{\epsilon}$ will tend to 1 and thereby indicate relative lack of structure.

The temporal orientation measurements, (3), can be taken as providing measures of the signal energy along the specified directions, $\hat{\mathbf{n}}$. This interpretation is justified by Parseval's theorem [38], which states that the sum of the squared values over the spacetime domain is proportional to the sum of the squared magnitude of the Fourier components over the frequency domain. Thus, for every spacetime location, $\mathbf{x}$, the local temporal energy $E_{\mathrm{T}}(\mathbf{x}; \hat{\mathbf{n}}_k, \sigma_j)$ measures the power of local temporal structure along each considered orientation $\hat{\mathbf{n}}_k$ and scale σ_j.

Figure 3 shows the temporal energies for the employed filter orientations on a sequence of the windmill farm. It is seen that the energies indicate dynamic information, *e.g.*, the dominant energies in Fig. 3(p)



(a) (b) $\theta_1 = 0°$ (c) $\theta_2 = 45°$ (d) $\theta_3 = 90°$ (e) $\theta_4 = 135°$ (f) $\theta_5 = 180°$ (g) $\theta_6 = 225°$ (h) $\theta_7 = 270°$ (i) $\theta_8 = 315°$

(j) $\hat{\mathbf{n}}_s = (0,0,1)^\top$ (k) $\hat{\mathbf{n}}_r = (1,0,1)^\top$ (l) $\hat{\mathbf{n}}_d = (0,-1,1)^\top$ (m) $\hat{\mathbf{n}}_l = (-1,0,1)^\top$ (n) $\hat{\mathbf{n}}_u = (0,1,1)^\top$ (o) $\hat{\mathbf{n}}_{hf} = (1,0,0)^\top$ (p) $\hat{\mathbf{n}}_{vf} = (0,1,0)^\top$ (q) $\hat{E}^{\epsilon}_{\mathrm{T}}$

Fig. 3: Spatial (first row) and temporal (second row) primitives for one temporal slice of a Windmill sequence (a) from the YUPENN dataset. (b)-(i) illustrate the spatial filtering results (1) for eight orientations θ. (j)-(q) illustrate the dynamic energies, (4), for the following directions $\hat{\mathbf{n}} = (n_x, n_y, n_t)^\top$: static (j), rightward (k), downward (l), leftward (m), upward (n), horizontal flicker (o), and vertical flicker (p). Further, (q) illustrates the no structure channel, (5). The hottest colors (*i.e.*, red) indicate the largest responses across each frame.

capture the movement of the rotor blades, as well as static temporal information with the energies shown in Fig. 3(j), reaching high values for the static structures on the ground of the scene. In contrast, Fig. 3(q) shows $\hat{E}^{\epsilon}_{\mathrm{T}}(\mathbf{x};\sigma_j)$, where large responses are seen in the unstructured sky region.

It is interesting to note that while the employed temporal and spatial features both make use of Gaussian derivatives to capture local orientation information, they differ in two ways. First, the spatial features, (1), rely on lower order derivatives in comparison to the temporal features, (2). In general, higher-order derivatives offer more precise orientation tuning than lower-order; however, for numerically stable estimates they require concomitantly larger support in implementation, which decreases precision in (spatiotemporal) localization [36]. In application to dynamic scenes, preliminary experiments showed that in this inevitable trade-off, precision in orientation was most important for temporal feature description (*i.e.*, direction of motion more important than exact position of motion), whereas precision in localization was most important for spatial feature description (*i.e.*, position more important than directionality). Second, the spatial feature measurements, (1), are not converted to (normalized) energies, while the temporal are, (4), because maintaining the contrast information is critical for adequately capturing spatial appearance.

3.1.3 Chromatic information

Previous evaluations [4, 5, 23, 24] showed that integrating color cues is useful for dynamic scene categorization. Color information is incorporated in the present spacetime primitives via the addition of three locally aggregated color measurements corresponding to the mean

$$\mu_m(\mathbf{x};\sigma_j) = \frac{1}{|\Omega|}\sum_{\Omega}\mathcal{I}_m(\mathbf{x}), \tag{6}$$

and the variance

$$\sigma_m^2(\mathbf{x};\sigma_j) = \frac{1}{|\Omega|}\sum_{\Omega}\left[\mathcal{I}_m(\mathbf{x}) - \mu_m(\mathbf{x};\sigma_j)\right]^2, \tag{7}$$

of the three CIE-LUV color channels, *i.e.* $m \in \{L, U, V\}$, and all other notation is by analogy with the filtering formula (1). Previous work on dynamic scene recognition has used simple color histograms to capture chromatic information [4, 5, 23, 24]. Here, however, following on other research that showed improved performance when moving from histogram to mean and variance based color representations [39], the latter approach is incorporated.

3.1.4 Temporal slice-based feature aggregation

The complementary spacetime orientation measurements presented so far are defined locally (pointwise) across a video sequence. To create descriptors that capture their surrounding regions, the feature descriptors are aggregated both temporally and spatially.

Initial temporal aggregation is performed by processing input video in discrete batches of Δt contiguous frames, referred to as *temporal slices*. Temporal slicing is motivated by the desire for incremental processing that can allow for efficient, on-line operation. Use of short-term parceling of the measurements

also is well matched with the restriction to use of fine temporal scale during spatiotemporal filtering to favor scene over camera dynamics.

Within each temporal slice, the local feature measurements are sampled with a spatial stride of $\Delta\mathbf{x}$ by centering patches of size $r_x \times r_y$ pixels and cuboids of size $r_x \times r_y \times r_t$, for spatial and temporal features, respectively. The neighborhood structure of these regions is further captured by subdividing them into $c_x \times c_y$ and $c_x \times c_y \times c_t$ sub-regions over which local feature measurements are aggregated into histograms. Here, the aggregation is naturally realized by setting the support of Ω for the spatial (1), temporal (3), and chromatic (6, 7), measurements to that of the sub-regions. Finally, for each sample point, $\mathbf{x}$, feature vectors, $\mathbf{v}_{\mathrm{S}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{T}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{C}}(\mathbf{x})$, are defined for the complementary spatial, temporal and chromatic measurements. These vectors are generated by concatenating the measurements for each of the sub-regions. Thus, each sample point is characterized by its complementary features and neighborhood.

During subsequent encoding and pooling stages of the processing pipeline, each temporal slice is treated individually, as described in the next two sections.

3.2 Feature encoding

A variety of different coding procedures exist to convert primitive local features, $\mathbf{v}(\mathbf{x}) \in \mathbb{R}^D$, into more effective intermediate-level representations, $\mathbf{c}(\mathbf{x})$, and the choice can significantly impact performance [10]. In the present case, the primitive features are given in terms of the complementary features, $\mathbf{v}_{\mathrm{S}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{T}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{C}}(\mathbf{x})$, defined in the previous section. For feature encoding, the present work considers two particularly effective coding techniques for dynamic scene classification, Locality-constrained Linear Coding (LLC) and Improved Fisher Vector (IFV); see [24] for a systematic empirical evaluation of a representative set of four contemporary coding techniques where LLC and IFV performed best in application to dynamic scenes. In both cases, the encoding is performed with respect to an unsupervised trained codebook $\mathbf{B} \in \mathbb{R}^{D \times K}$.

In general, largely two categories of improved encoding approaches exist in the literature. One category expresses features as combinations of (sparse) codewords (*e.g.*, [40, 41, 42]). The other category considers differences between the original features and the codewords (*e.g.*, [39, 43, 44, 45]). LLC [41] is a particularly strong performer of the first category. The LLC code $\mathbf{c}_{\mathrm{LLC}}(\mathbf{x}) \in \mathbb{R}^K$ encodes each local feature $\mathbf{v}(\mathbf{x})$ by the $M \ll K$ nearest codewords in $\mathbf{B}$ which is trained by quantizing the extracted descriptors from training sequences with K-means. Fisher vectors (FV) [43] are considered as a representative of the second category, as they typically outperform alternatives [10]. An FV $\mathbf{c}_{\mathrm{FV}}(\mathbf{x}) \in \mathbb{R}^{2KD}$ models mean and covariance gradients between a set of features $\left\{\mathbf{v}(\mathbf{x}) \in \mathbb{R}^D\right\}$ and the components of a codebook which is learned on training descriptors by using a Gaussian mixture model (GMM). The Improved Fisher Vector (IFV), $\mathbf{c}_{\mathrm{IFV}}(\mathbf{x})$, is obtained by computing the signed square-root to each element of $\mathbf{c}_{\mathrm{FV}}(\mathbf{x})$ followed by ℓ_2 normalization [39]. The present work only focuses on the improved version of the Fisher vector, since it consequently outperforms FV when coupled with linear SVM classifiers [24, 39].

Note that computing the average first and second order differences between the features and each of the GMM centres implicitly performs an average pooling of the local features in the Fisher vector representation. LLC codes on the other hand are best pooled via max-pooling, which takes the strongest codeword response in a region and has been shown to be more discriminative than average pooling [41, 46].

3.3 Dynamic feature pooling

When pooling the encoded features, $\mathbf{c}(\mathbf{x})$, from dynamic scenes, those that significantly change their spatial location across time should be pooled adaptively in a correspondingly dynamic fashion. For example, global image motion induced by a camera pan could cause the image features to move with time and pooling that is tied to finely specified image location will fail to capture this state of affairs. Similarly, when regions change their spatial relations with time, pooling should adapt. In such situations, a lack of appropriately dynamic pooling will degrade recognition performance, as features pooled at one location will have moved to a different location at a subsequent time and thereby be at risk of improper matching. Significantly, this challenge persists if the pooling positions are hierarchically arranged [16] or even more adaptively defined [42, 47, 48], but without explicit attention to temporal changes in pooling regions.

In contrast, features that retain their image positions over time (*i.e.*, static patterns) can be pooled within finer, predefined grids, *e.g.*, as with standard spatial pyramid matching (SPM) [16]. Indeed, even highly dynamic features that retain their overall spatial position across time (*i.e.*, temporally stochastic patterns, such as fluttering leaves on a bush and other dynamic textures) can be pooled with fine grids. Thus, it is not simply the presence of image dynamics that should relax finely gridded pooling, but rather the presence of larger scale coherent motion (*e.g.*, as encountered with global camera motion).

3.3.1 Dynamic pooling energies

In response to the above observations, a set of dynamic energies have been derived that favor orderless pooling (*e.g.*, global aggregation) when coarse scale image motion dominates and spatial pooling (as in an SPM scheme) when an encoded feature is static or

its motion is stochastic but otherwise not changing in overall spatial position. These energies are used as pooling weights applied to the locally encoded features so that they can be pooled in an appropriate fashion.

The directional spacetime energies, (3), so far derived are not sufficient for distinguishing between so called coherent motion (*e.g.*, as exemplified by large scale motion resulting from camera movement) and incoherent motion (*e.g.*, as exemplified by stochastic dynamic textures) [49, 50]. In the present context, suppose that the directional energies are recovered along the leftward, rightward, upward, downward and four diagonal directions as well as static (zero velocity), which will be denoted as $l, r, u, d, ru, rd, lu, ld$ and s, respectively. The desired pooling energies are meant to capture coherent motion and that goal can be accomplished by combining the directional energies in opponent-motion channels as follows

$$\begin{aligned} E^{\mathcal{P}}_{|r-l|}(\mathbf{x};\sigma_j) &= |E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_r,\sigma_j) - E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_l,\sigma_j)| \\ E^{\mathcal{P}}_{|u-d|}(\mathbf{x};\sigma_j) &= |E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_u,\sigma_j) - E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_d,\sigma_j)| \\ E^{\mathcal{P}}_{|ru-ld|}(\mathbf{x};\sigma_j) &= |E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_{ru},\sigma_j) - E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_{ld},\sigma_j)| \\ E^{\mathcal{P}}_{|lu-rd|}(\mathbf{x};\sigma_j) &= |E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_{lu},\sigma_j) - E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_{rd},\sigma_j)| \end{aligned} \quad (8)$$

to yield a set of dynamic pooling energies, $E^{\mathcal{P}}$, representing coherent image motion in 4 equally spaced directions (horizontal ($r-l$), vertical ($u-d$) and two diagonals ($ru-ld$ and $lu-rd$)).

In contrast to the individual motion direction consistent energies, (3), the pooling energies, (8), explicitly capture coherent motion across various directions. For example, a spatial region with a stochastically moving spacetime pattern, *e.g.* the leaves of a tree in the wind can exhibit large motions in several specific directions $\hat{\mathbf{n}}$; however, after taking the absolute arithmetic difference from opponent directions, the coherent motion pooling energies, (8), of such stochastic spacetime texture patterns are approximately zero. On the other hand, regions that are dominated by a single direction of motion (*i.e.* coherent motion regions) will yield a large response in the most closely matched channel. (For an extended discussion of the relationship between spatiotemporal oriented energies and coherent motion see, *e.g.* [49].)

The pooling energies so far defined, (8), are ℓ_1 normalized together with the static energy channel, $E_{\mathrm{T}}(\mathbf{x};\hat{\mathbf{n}}_s,\sigma_j)$, that indicates lack of coarse motion

$$\hat{E}^{\mathcal{P}}_k(\mathbf{x};\sigma_j) = \frac{E^{\mathcal{P}}_k(\mathbf{x};\sigma_j)}{\sum_{\lambda\in\Lambda} E^{\mathcal{P}}_\lambda(\mathbf{x};\sigma_j)+\epsilon}, \forall k \in \Lambda, \quad (9)$$

to yield a point-wise distribution of static, coherent, as well as unstructured energy via the normalized ϵ indicating homogeneous regions

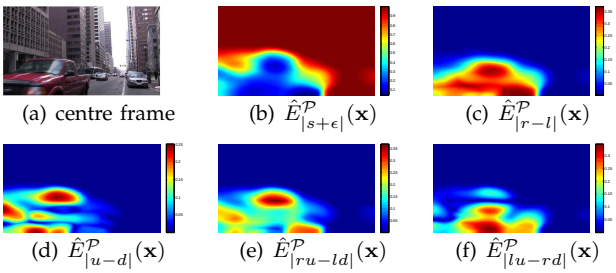
$$\hat{E}^{\mathcal{P}}_\epsilon(\mathbf{x};\sigma_j) = \frac{\epsilon}{\sum_{\lambda\in\Lambda} E^{\mathcal{P}}_\lambda(\mathbf{x};\sigma_j)+\epsilon}, \quad (10)$$

with $\Lambda = \{s, |r-l|, |u-d|, |ru-ld|, |lu-rd|\}$. Further, since regions without coherent motion or with only fine scale motion (indicated by $\hat{E}^{\mathcal{P}}_s$), as well as homogeneous regions (indicated by $\hat{E}^{\mathcal{P}}_\epsilon$), can be similarly pooled with spatial gridding to capture geometric layout, static energy is summed with unstructured energy as

$$\hat{E}^{\mathcal{P}}_{|s+\epsilon|}(\mathbf{x};\sigma_j) = \hat{E}^{\mathcal{P}}_s(\mathbf{x};\sigma_j) + \hat{E}^{\mathcal{P}}_\epsilon(\mathbf{x};\sigma_j), \quad (11)$$

to yield the final set of (coherent) motion directions, $\Lambda = \{s+\epsilon, |r-l|, |u-d|, |ru-ld|, |lu-rd|\}$, that specify the dynamic pooling energies.

The dynamic pooling energies for a temporal subset of a street sequence are shown in Fig. 4. Figure 4(a) depicts the central frame of the filtered sequence and 4(b)-4(f) show the decomposition of the filtered sequence into a distribution of static and coherent motion dynamic energies. Observe that the static+unstructured channel consists of large responses for stationary image structures, *e.g.*, the buildings in the scene, as well as for homogeneous regions such as the sky in the center of the scene. Whereas the foreground red car's dynamic energy can be decomposed into several coherent motion channels with a large part originating from the horizontal motion channel, *i.e.*, $\hat{E}^{\mathcal{P}}_{|r-l|}(\mathbf{x})$, shown in Figure 4(c). Note that fine-scale motions, such as the moving cars in the background, are not captured by the coherent motion channels (Fig. 4(c)-4(f)) and therefore exhibit strong responses in the static channel 4(b), which is appropriate as they form (part of) the background dynamic texture.



(a) centre frame (b) $\hat{E}^{\mathcal{P}}_{|s+\epsilon|}(\mathbf{x})$ (c) $\hat{E}^{\mathcal{P}}_{|r-l|}(\mathbf{x})$

(d) $\hat{E}^{\mathcal{P}}_{|u-d|}(\mathbf{x})$ (e) $\hat{E}^{\mathcal{P}}_{|ru-ld|}(\mathbf{x})$ (f) $\hat{E}^{\mathcal{P}}_{|lu-rd|}(\mathbf{x})$

Fig. 4: Distribution of Dynamic Pooling Energies of a Street Sequence from the YUPENN Dataset. (b)-(f) show the decomposition of the sequence into a distribution of pooling energies indicating stationarity/homogeneity in (b) and coarse coherent motion for several directions in (c)-(f). Hotter colors (*e.g.*, red) correspond to larger filter responses.

3.3.2 Dynamic spacetime pyramid

The pooling process creates a non-local representation by collecting the local feature codes in spatial subregions based on summing (average-pooling) or taking the maximum (max-pooling) across the codes. When pooling the encodings, all analyzed feature

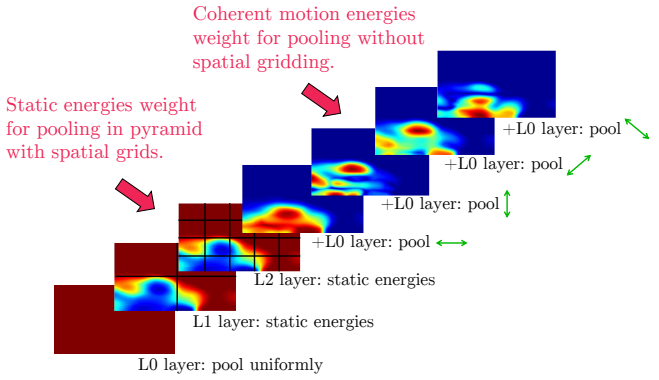


Fig. 5: Dynamic Spacetime Pyramid. Larger responses (hotter colors) in the static energies guide pooling into coarse (L0) as well as fine grids (L1 and L2). In contrast, coherent motion energies limit pooling to the coarsest grid only (L0). Green arrows along the sides of the coherent motion plots indicate directional tunings. Input imagery is from the Street Sequence, as shown in Fig. 4.

representations use a three-level spatial pyramid [16] to capture weak spatial information in the feature vectors, as this has previously been shown to increase recognition accuracy [24]. The feature encodings are pooled therein and concatenated for the final feature vector. Thus, the basic pooling architecture partitions the image into subdivisions constructed for each sample point in time, *i.e.* each temporal slice is partitioned into 1×1, 2×2, and 4×4 spatial subdivisions, resulting in 21 pyramid channels.

The proposed dynamic pyramid extends the standard three-level spatial pyramid [16] as follows. For pooling at the coarsest pyramid level $l = 0$, *i.e.*, in the region without spatial grid, feature codes are aggregated globally. In regions with spatial grids, however, *i.e.*, $l > 0$ the static pooling energies $\hat{E}^{\mathcal{P}}_{|s+\epsilon|}(\mathbf{x})$ are used as weights, emphasizing the local contribution of each visual word. Lastly, to explicitly pool features favorably from regions with coherent motion, four more channels $\Lambda = \{|r-l|, |u-d|, |ru-ld|, |lu-rd|\}$, which perform weighted pooling with coherent energies $\hat{E}^{\mathcal{P}}_{\lambda}(\mathbf{x})$, are added to the pyramid. Due to the coarse-scale motion of these features, the top pyramid level $l = 0$ without spatial information is used. Therefore, the final spatiotemporal pyramid encodes a sample point in time in 25 channels, with each channel capturing specific spatial and temporal properties of the pooled codewords. See Fig. 5 for an illustrative example.

Significantly, since all dynamic pooling energies are normalized jointly, (9) and (10), the pooling has exactly the desired effect: Feature codes derived from points with larger magnitude static energies, $\hat{E}^{\mathcal{P}}_{|s+\epsilon|}$, are preferentially pooled across all available spatial griddings to capture spatial layout. In contrast, feature codes derived from points with larger magnitude coherent motion energies receive small weights from the static energies during finer grid pooling and will instead be preferentially pooled globally according to their particular motion direction; thus, the overall movement of such features is captured without contaminating spatial layout information. For example, encoded features on horizontally moving objects are pooled with high corresponding weights $\hat{E}^{\mathcal{P}}_{|r-l|}$ to explicitly capture horizontally moving image structures in the dynamic spacetime pyramid.

This novel pooling process based on dynamic weights is independent of the underlying type of pooling. For average pooling, the dynamic energies are used as weights in the code-summation process; in particular, when computing the mean and covariance gradients of the Fisher vector representation, the local image features are weighted multiplicatively with the dynamic energies. Similarly, for max-pooling used in LLC representations the location of the strongest codeword is found following multiplicative weighting of the codes with the dynamic pooling energies.

Finally, a global feature vector, $\mathbf{f}$, representing a point in time of the video, is constructed by concatenating the feature poolings of all pyramid channels. These feature vectors then serve as the basis of an online recognition scheme when coupled with a classifier, *e.g.*, a Support Vector Machine (SVM).

3.4 Detailed implementation summary

1) **Local feature extraction.** The video is processed in a temporal sliding window by dense extraction of spatial (1), temporal (3) and chromatic (6, 7) primitives at every $\Delta t = 16$ frames (the duration of a temporal slice is 16 frames). All features are extracted at $|\sigma| = 5$ different scales by downscaling the image (*i.e.* spatial domain only in preference for capturing short term temporal variations) by factors of $\sqrt{2}$. For the oriented spatial and spatiotemporal filtering operations, (1) and (2), the filter lengths are set to 5 and 13 pixels and the number of filter orientations are set to $|\theta| = 2$ and $|\theta| = 10$ in order to span the orientation space for the order and dimensionality of filters employed [51]. The final descriptors, $\mathbf{v}_{\mathrm{S}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{T}}(\mathbf{x})$, $\mathbf{v}_{\mathrm{C}}(\mathbf{x})$ are extracted with a spatial stride of $\Delta\mathbf{x} = 8$ pixels and cover regions of size $r_x \times r_y \times r_t = 16 \times 16 \times 16$ which are divided into $c_x \times c_y \times c_t = 2 \times 2 \times 3$ sub-regions for histogram summation. To construct $\mathbf{v}_{\mathrm{S}}(\mathbf{x})$, the spatial orientations for steering the basis filter responses is set to 8, creating $D_{\mathrm{S}} = 2 \times 2 \times 8 = 32$ dimensional spatial descriptors. The temporal descriptors, $\mathbf{v}_{\mathrm{T}}(\mathbf{x})$, capture 8 directions, parameterized by $\hat{\mathbf{n}}$ corresponding to motion along the leftward, rightward, upward and downward directions as well as static (zero velocity), flicker (infinite horizontal / vertical velocity) and unstructuredness ((5) with $\epsilon = 500$), to result in $D_{\mathrm{T}} = 2 \times 2 \times 3 \times 8 = 96$ dimensions. Lastly, the color

descriptors, $\mathbf{v}_{\mathrm{C}}(\mathbf{x})$, record LUV mean and variance of the sub-regions in $D_{\mathrm{C}} = 2 \times 2 \times 6 = 24$ dimensions. As a post-processing step, RootSIFT normalization [52] is applied to each descriptor *i.e.*, signed square rooting each dimension followed by ℓ_2 normalization. Notably, owing to the separability and steerabilty of the underlying filtering operations, all features can be extracted with modest computational expense.

2) **Feature encoding.** The local descriptors are encoded either via LLC or IFV. A random subset of features from the training set, consisting of a maximum of 1000 descriptors from each training sequence, are used to learn a visual vocabulary. For LLC, the codebook entries are learned by quantizing the extracted descriptors from the training sequences with K-means to $K = 200$ codewords. All parameters in LLC are set to the default values from the original publications [10, 41]. To maintain low computational complexity, an approximate nearest neighbour search is used for efficient clustering and encoding. In the case of Fisher vectors, a GMM with $K_{\mathrm{S}} = 50$, $K_{\mathrm{T}} = 100$, $K_{\mathrm{C}} = 10$ mixtures is fit to the subsampled training descriptors. As shown in Section 4, different codebook sizes can impact performance. Before IFV encoding, PCA whitening is applied to the descriptors to reduce their dimension by a factor of two. Data decorrelation via PCA also supports the diagonal covariance assumptions in the employed GMM [45].

3) **Feature pooling.** To compare against conventional pooling, an $l = 3$ level SPM is used to maintain weak spatial information of the features extracted in each temporal instance. The resulting 21 pooling regions from spatial grids of size $2^l \times 2^l$ create a $21 \times K = 21 \times 200 = 4200$ dimensional feature vector for LLC encoding and a $21 \times 2 \times K_{\mathrm{GMM}} \times D = 21 \times 2 \times K_{\mathrm{GMM}} \times 28 = 1176 \times K_{\mathrm{GMM}}$ dimensional feature vector for the Fisher encoding. As in the original publications, pooling is performed using either the maximum of the encoded features (LLC) or summation of the Fisher vector gradient (IFV) over the pooling region. To calculate the proposed dynamic pooling energies, the spatiotemporal aggregation region Ω in (2) is set spatially to one fourth of the video frame size and temporally to the length of the spatiotemporal filters (13 frames).

4) **Learning and Classification.** Each set of encoded features pooled from the same temporal instance generates a feature vector, $\mathbf{f}$. For training, all feature vectors extracted from the training set are used to train one-vs-rest linear SVM classifiers with ℓ_2 normalization applied to the feature vectors. The SVM's regularization loss trade-off parameter is set to $C = 1$. During classification, each feature vector for each temporal slice of a video is classified by the one-vs-rest SVM to yield a temporal prediction; overall classification for each feature type is according to the majority of the predictions of each slice. Preliminary experiments that instead immediately classified across the entire video revealed lower classifcation performance than classifying initially by temporal slices. Temporal slicing increases the training data by a large factor and allows the videos to be classified very early, since the classifications of the temporal slices are mostly equal over time and vary only in cases with large temporal changes, e.g., after scene cuts. The complementary features are combined by late fusion of their respective SVM scores, which are linearly combined with weights determined after cross-validation on the training data, to yield a final classification of a video.

4 Empirical Evaluation

The proposed Dynamically Pooled Complimentary Feature (DPCF) system is evaluated on the Maryland "In-The-Wild" [4] and YUPENN [5] dynamic scene recognition datasets. The datasets contain videos showing a wide range of natural dynamic scenes (avalanches, traffic, forest fires, waterfalls, *etc.*); see Tables 3 and 4 where complete listings can be read off the left most columns of the tables. It should be noted that the Maryland dataset contains a high degree of coarse scale motion (mostly from camera movement), whereas the YUPENN dynamic scene sequences are captured from static cameras only.

A leave-one-video-out experiment is used for consistency with previous evaluations [4, 5, 23, 24, 27]. The structure of the experiments is two-layered. First, Section 4.1 evaluates the proposed complementary primitive features of Section 3.1. This evaluation includes LLC feature encoding that is based on local codeword statistics and IFV encoding based on the difference between the codewords and features to encode. The section also evaluates the novel dynamic pooling framework of Section 3.3. Second, in Section 4.2 the full proposed complementary system is compared with the state-of-the-art in dynamic scene classification.

4.1 Feature comparison and combination

In Table 1, the average classification performance is shown for the three proposed complementary feature primitives, the two investigated coding approaches as well as the proposed dynamic pooling.

The results comparing the three descriptor types show that the spatial descriptors, $\mathbf{v}_{\mathrm{S}}$, perform best on the Maryland dataset which, to some degree, can be attributed to the highly divergent motion information (even for videos of the same class). On the other hand, the temporal descriptor, $\mathbf{v}_{\mathrm{T}}$, achieves overall best classification rates on the YUPENN dataset.

Color descriptors, $\mathbf{v}_{\mathrm{C}}$, yield considerably worse performance, being around 20% lower than their spatial and temporal complements. As noted in Section 3.4, the late fusion of (*all*) three descriptors is carried out by combining the SVM score vectors with weights

Maryland "In-The-Wild" dataset																
Primitives	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_C$	$\mathbf{v}_C$	$\mathbf{v}_C$	$\mathbf{v}_C$	all	all	all	all
Encoding	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV
Pooling	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.
Avg. Perf.	67.69	69.23	74.62	**77.69**	59.23	60.00	65.38	70.77	49.23	52.31	54.62	55.38	70.00	75.38	72.31	**80.00**

YUPENN Dynamic Scenes dataset																
Primitives	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_S$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_T$	$\mathbf{v}_C$	$\mathbf{v}_C$	$\mathbf{v}_C$	$\mathbf{v}_C$	all	all	all	all
Encoding	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV	LLC	LLC	IFV	IFV
Pooling	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.	static	dyn.
Avg. Perf.	89.52	91.67	91.90	94.76	94.52	95.00	97.38	**97.62**	68.81	70.71	77.38	79.05	96.43	96.90	96.90	**98.81**

TABLE 1: Results for combinations of feature primitives, encodings and pooling. The average recognition accuracy in % for classification with one vs. rest linear SVMs is reported.

estimated by cross-validation on the training set. The results indicate that the proposed descriptors are not only complementary in the sense of their design, but also in the way that their combination further improves over the best single feature performance.

When comparing the different feature encodings on the two datasets it is seen that while both the LLC and IFV approaches provide good overall performance, the higher-order IFVs consistently outperform the lower dimensional LLC encoding. The benefit is particularly prominent on the "In-The-Wild" dataset; apparently the higher dimensional encoding is especially advantageous on data with a high degree of intra-class variations, as *e.g.* introduced by significant camera motion.

Table 1 also compares *static* pooling within a standard three-level spatial pyramid [24] against the proposed dynamic spacetime pyramid (*dyn.*). Note that this comparison is independent of the underlying pooling principle, being either conventional max- (LLC) or average-pooling (IFV). One observes that dynamic pooling within the proposed dynamic spacetime pyramid leads to best performance on both datasets. Conventional pooling strategies are outperformed by a margin of 7.7% and 1.9% for Maryland and YUPENN, respectively. Note that dynamic pooling always (also for spatial features) increases performance on the stabilized YUPENN dataset where all observable motion is due to scene dynamics. This result empirically verifies that, when pooling with a spatial grid, giving low weights to features with coarse scale movement facilitates recognition performance.

The significant performance gain associated with dynamic pooling on the Maryland dataset can be attributed to the significant temporal variations and the severe camera movement that is present in the videos of this dataset. Since camera movement generally manifests itself at coarse temporal scales and the proposed dynamic pooling method favors pooling without geometric context within the dynamic pyramid when coarse (coherent) motion is present, it avoids inappropriate spatially gridded pooling when image structure drastically changes its position with time. The approach thereby becomes robust to camera (and other coarse) motions.

LLC encoding						
K	50	100	200	400	600	800
dimension	1250	2500	5000	10000	25000	40000
Maryland dataset						
$\mathbf{v}_S$	60.77	66.15	69.23	73.08	73.85	73.85
$\mathbf{v}_T$	56.92	56.92	60.00	58.46	61.54	60.77
$\mathbf{v}_C$	50.00	51.54	52.31	52.31	50.00	53.08
YUPENN dataset						
$\mathbf{v}_S$	89.05	90.71	91.67	91.76	92.38	91.90
$\mathbf{v}_T$	89.52	92.62	95.00	95.48	93.33	93.57
$\mathbf{v}_C$	67.14	69.29	70.71	73.10	73.10	73.10
IFV encoding						
K_{GMM}	10	20	50	100	150	250
dimension	11760	23520	58800	117600	176400	294000
Maryland dataset						
$\mathbf{v}_S$	73.85	75.38	77.69	76.15	76.15	76.15
$\mathbf{v}_T$	59.23	62.31	67.69	70.77	69.23	66.92
$\mathbf{v}_C$	57.69	55.38	53.08	54.62	50.77	53.08
YUPENN dataset						
$\mathbf{v}_S$	94.29	95.48	94.76	95.00	94.76	94.29
$\mathbf{v}_T$	98.33	97.14	97.38	97.62	97.38	97.14
$\mathbf{v}_C$	80.00	78.33	76.19	75.24	74.52	72.62

TABLE 2: Overall classification accuracy for different codebook sizes when using LLC and IFV encoded features pooled via the proposed dynamic pooling framework.

Consideration of the YUPENN results shows that the dynamic pooling advantage is had without compromising performance when camera motion is absent. Here, the dynamic pooling allows aggregation at finer levels of the dynamic pyramid to more precisely localize the spatiotemporal image structure. Interestingly, there is even a slight improvement on YUPENN under dynamic pooling, which may be due to the fact that coherently moving objects are specifically matched by the dynamic pooling channels. For example, vertically moving visual codes from a waterfall sequence will be explicitly matched, since these are favorably pooled within the $\hat{E}^{\mathcal{R}}_{|u-d|}$ channel of the dynamic spacetime pyramid.

Varying the size of the codebook can impact classifi-

Class	HOF [9]+ GIST [13]	Chaos [4]+ GIST [13]	SOE [5]	SFA [27]	CSO [23]	BoSE [24]	C3D [53]	DPCF $\{\mathbf{v}_S, \mathbf{v}_T\}$	CF $\{\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C\}$	DPCF $\{\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C\}$
Avalanche	20	60	40	60	60	60	90	80	80	90
Boiling Water	50	60	50	70	80	70	90	80	60	60
Chaotic Traffic	30	70	60	80	90	90	90	100	90	100
Forest Fire	50	60	10	10	80	90	80	80	60	90
Fountain	20	60	50	50	80	70	60	60	70	80
Iceberg Collapse	20	50	40	60	60	60	60	50	50	50
Landslide	20	30	20	60	30	60	70	80	60	80
Smooth Traffic	30	50	30	50	50	70	80	80	70	70
Tornado	40	80	70	70	80	90	80	80	80	80
Volcanic Eruption	20	70	10	80	70	80	90	90	90	90
Waterfall	20	40	60	50	50	100	40	70	60	70
Waves	80	80	50	60	80	90	100	100	100	100
Whirlpool	30	50	70	80	70	80	80	70	70	80
Overall	33	58	43	60	68	78	78	78	72	**80**

TABLE 3: Classification accuracy (in %) for the best performing approaches on the Maryland dataset.

Class	HOF [9]+ GIST [13]	Chaos [4]+ GIST [13]	SOE [5]	SFA [27]	CSO [23]	BoSE [24]	C3D [53]	DPCF $\{\mathbf{v}_S, \mathbf{v}_T\}$	CF $\{\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C\}$	DPCF $\{\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C\}$
Beach	87	30	93	93	100	100	97	100	93	100
Elevator	87	47	100	97	100	97	100	100	100	100
Forest Fire	63	17	67	70	83	93	100	93	93	97
Fountain	43	3	43	57	47	87	83	96	93	93
Highway	47	23	70	93	73	100	97	100	100	100
Lightning Storm	63	37	77	87	93	97	93	100	100	100
Ocean	97	43	100	100	90	100	100	100	100	100
Railway	83	7	80	93	93	100	97	100	100	100
Rushing River	77	10	93	87	97	97	100	100	100	100
Sky-Clouds	87	47	83	93	100	97	97	100	100	100
Snowing	47	10	87	70	57	97	93	97	97	97
Street	77	17	90	97	97	100	100	100	97	100
Waterfall	47	10	63	73	77	83	97	93	87	97
Windmill Farm	53	17	83	87	93	100	100	97	97	100
Overall	68	23	81	85	86	96	97	98	97	**99**

TABLE 4: Classification accuracy (in %) for the best performing approaches on the YUPENN dataset.

cation performance, as shown in Table 2. The number of visual words, K, that represents the number of centroids in the LLC representation, and K_{GMM}, which denotes the number of mixtures used in the GMM for Fisher vectors, is varied. The resulting dimension of the feature vector for a single temporal slice is listed as well. When increasing the codebook size, performance increases up to a certain point. Generally, a small vocabulary size decreases discriminability between the classes. In contrast, a large vocabulary makes it difficult to find similar codewords within instances of the same class, as features describing similar visual input will be mapped to different codewords. This point explains the performance decrease for larger codebooks on the Maryland dataset, because it exhibits higher intra-class variations than YUPENN.

4.2 Comparison with the state-of-the-art

The proposed approach is compared to several others that previously have shown best performance in dynamic scene recognition: GIST [13] + histograms of flow (HOF) [9], GIST + chaotic dynamic features (Chaos) [4], spatiotemporal oriented energies (SOE) [5], slow feature analysis (SFA) [27], 3D ConvNet (C3D) features [53], and previous work by the authors, complementary spacetime orientation (CSO) features [23] and Bags of Spacetime Energies (BoSE) [24].

The principal DPCF representation is the best performing variation considered in Section 4.1: densely extracted complementary descriptors (spatial $\mathbf{v}_S$, temporal $\mathbf{v}_T$, and color $\mathbf{v}_C$) that are encoded by IFV using the dynamic pyramid representation. All parameter choices are given in Section 3.4. For the sake of further comparison, also considered are variations of the proposed approach that consider only spatial, $\mathbf{v}_S$, and temporal, $\mathbf{v}_t$, features and for the full set of complementary features (*i.e.* also including $\mathbf{v}_c$), but using standard (3-level) spatial pyramid pooling, i.e., without the dynamic pooling of Section 3.3. Results are shown in Tables 3 (Maryland dataset) and 4 (YUPENN dataset).

For both datasets, the complete DPCF approach ($\{\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C\}$) achieves a new state-of-the-art in outperforming the previous best performers, BoSE [24] and C3D [53], which were essentially on par. On the Maryland dataset, the novel DPCF representation achieves an average accuracy of 80% when coupled with a simple linear SVM classifier. The proposed

approach's 99% accuracy on YUPENN shows that performance is saturated on this dataset. On a single class-level, outstanding results are the high accuracies for the Fountain class, which exhibits huge intra-class variations in the background and only a small amount of common foreground (*i.e.*, the fountain itself).

Improvement over BoSE and CSO underlines the advances of the current approach in comparison to previous efforts by the authors. Compared to CSO [23], DPCF increases by 12% and 13% on YUPENN and Maryland. This underlines the importance of aggregating the primitive features in grids so as to capture spatiotemporal neighborhood structure as well as a local encoding and dynamic pooling strategy. Moreover, DPCF boosts performance of BoSE by by 2% and 3% on YUPENN and Maryland which underlines the importance of encoding complementary spatial, temporal and color cues in an IFV representation by using a dynamic aggregation strategy (Section 3.3).

Improvement over C3D shows the merit of the approach in comparison to the strongest performing application of deep convolutional networks to 3D spatiotemporal image processing for scene recognition, even while maintaining lower complexity, single level feature extraction.

Interestingly, while the inclusion of color features in DPCF yields modest overall improved performance vs. using only spatial and temporal features, $\{\mathbf{v}_S, \mathbf{v}_T\}$, the amount of improvement varies drastically by class. As examples: Forest Fire gains 10% on Maryland and 4% on YUPENN; apparently the colors of fire are distinctive. In contrast, scenes involving vehicular traffic, Highway on YUPENN and Chaotic as well as Smooth Traffic on Maryland, gain nothing and can even decrease in performance; apparently the particular color of cars in a given traffic scene is of little use and can even be confusing in class categorization.

Further useful observations can be made by comparing the results for the Complementary Features lacking dynamic pooling (CF) against the Dynamically Pooled Complementary Feature (DPCF). Here, it is seen that, especially on Maryland (Table 3), recognition rates increase for several classes, *e.g.*, Forest Fire and Landslide, which further underline the benefits of dynamic pooling in the presence of camera motion. Also, one again notices that for all classes on the stabilized YUPENN dataset the dynamic pooling increases recognition rates, most notably for the Waterfall class where the coherent motion channels of the dynamic spacetime pyramid cause an increase of 10% in accuracy. Improvements here arise owing to the ability of dynamic pooling to benefit recognition in situations where primitive components (*e.g.*, localized measure of water flow) move across time, even in otherwise static scenes.

Finally, it is interesting to compare performance in particular with respect to the stabilized vs. unstabilized camera natures of the YUPENN and Maryland datasets: For classes with the same label across datasets (*e.g.* Forest Fire, Fountain, Waterfall) as well as highly similar classes (*e.g.* Highway and Smooth Traffic), while benefits are had from dynamic pooling, performance remains compromised in the presence of unstabilized camera motion.

4.3 Class confusions

Confusion tables for the proposed approach are shown in Table 5. Again, the proposed DPCF representation consists of densely extracted complementary descriptors that are dynamically encoded by IFV using the proposed dynamic pyramid representation, parameter choices as given in Section 3.4. It can be observed that most of the confusions are between visually similar scene classes. For example, on Maryland, Smooth Traffic is confused with Chaotic Traffic. On YUPENN, confusions only occur between highly similar classes, *e.g.* Fountain and Waterfall, showing dynamic water textures.

5 Conclusion

This paper has presented a complete system for dynamic scene recognition that tackles video analysis in a principled and well-founded manner in three main steps: primitive feature extraction, feature encoding and dynamic pooling. The entire processing pipeline is structured by an explicit modeling of feature complementarity, distinguishing between spatial, temporal, and color primitives $(\mathbf{v}_S, \mathbf{v}_T, \mathbf{v}_C)$. The system has been thoroughly evaluated on the two current benchmark datasets for dynamic scene recognition and yields the currently best performance on both datasets. This performance gain is due to (a) careful design of the complementary features, which is supported by biological evidence in parvocellular, magnocellular and konio layers of natural visual systems, (b) careful selection of encoding techniques and (c) a novel dynamic pooling approach, which greatly increases performance when camera motion is present, without compromising performance when camera motion is absent.

A detailed analysis of the experimental results reveals that, on the complementary feature level, spatial primitive features, $\mathbf{v}_S$, perform best on videos with large camera motion (Maryland "In-The-Wild"), but temporal features, $\mathbf{v}_T$, are better on stabilized videos (YUPENN dataset). Significantly, however, the combination of all three complementary features outperforms each individual feature type, a result that underlines the complementarity of spatial appearance, temporal dynamics and color. At the encoding step, for all possible combinations of primitive features, Improved Fisher Vectors (IFV) consistently outperform Locality-constrained Linear Coding (LLC). Furthermore, it has been found that dynamic pooling can

	Avalanche	*Boiling Water*	*Chaotic Traffic*	*Forest Fire*	*Fountain*	*Iceberg Collapse*	*Landslide*	*Smooth Traffic*	*Tornado*	*Volcanic Eruption*	*Waterfall*	*Waves*	*Whirlpool*
Avalanche	**9**											1	
Boiling Water	1	**6**		1							1		1
Chaotic Traffic			**10**										
Forest Fire				**9**									1
Fountain	1		1		**8**								
Iceberg Collapse	1		1		1	**5**				1		1	
Landslide							**8**		1			1	
Smooth Traffic			2					**7**			1		
Tornado				1					**8**	1			
Volcanic Eruption						1				**9**			
Waterfall					2		1				**7**		
Waves												**10**	
Whirlpool	1		1										**8**

	Beach	*Elevator*	*Forest Fire*	*Fountain*	*Highway*	*Lightning St.*	*Ocean*	*Railway*	*Rushing River*	*Sky-Clouds*	*Snowing*	*Street*	*Waterfall*	*Windmill Farm*
Beach	**30**													
Elevator		**30**												
Forest Fire			**29**			1								
Fountain				**28**									2	
Highway					**30**									
Lightning St.						**30**								
Ocean							**30**							
Railway								**30**						
Rushing River									**30**					
Sky-Clouds										**30**				
Snowing											**29**		1	
Street												**30**		
Waterfall				1									**29**	
Windmill Farm														**30**

TABLE 5: Confusion Matrices for DPCF for the Maryland (left) and YUPENN (right) dataset. The columns show the predicted labels of the classifier, while the rows list the actual ground truth label.

adapt to global as well as local motion and always yields better results than static pooling.

Regarding the two benchmark datasets for dynamic scene recognition, the stabilized YUPENN dataset is better structured and balanced with respect to size and length of individual video clips than the less systematic video collection in the Maryland dataset. However, the Maryland dataset offers the additional challenge of camera motion. As expected, recognition rates are much better on stabilized videos, as compared to cases of significant camera motion. Indeed, an overall recognition rate of almost 99% shows that experimental validation on the YUPENN dataset has reached its limits. These observations emphasize the need for a new challenging dataset that contains more classes and more videos, while being well-structured and balanced with respect to the size and length of all individual video clips as well as to the cases of stationary and moving cameras.

The outstanding performance of the presented spacetime recognition framework on dynamic scenes suggests application to a variety of other areas, such as event retrieval and video indexing as well as object and activity localization. Indeed, examples suggesting the generality of the approach already are available. Strong performance in dynamic texture recognition was documented for an ancestor of the current approach [3] that relied on only temporal features, $\mathbf{v}_{\mathrm{T}}$, and pooling at the top level of a spatial pyramid for strictly global aggregation. Somewhat complementarily, an action recognition approach [54] has been presented that exploits more locally defined saliency-based dynamic pooling of $\mathbf{v}_{\mathrm{T}}$.

ACKNOWLEDGMENTS

This work was supported by the Austrian Science Fund (FWF) under project P27076 "Space-Time Representation and Recognition in Computer Vision" and a Canadian NSERC Discovery grant.

REFERENCES

[1] M. Szummer and R. Picard, "Temporal texture modeling," in *Proc. IEEE Int. Conf. Image Proc.*, 1998.

[2] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto, "Dynamic textures," *Int. J. Comput. Vis.*, vol. 51, no. 2, pp. 91–109, 2003.

[3] K. Derpanis and R. P. Wildes, "Spacetime texture representation and recognition based on a spatiotemporal orientation analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 6, pp. 1193–1205, 2012.

[4] N. Shroff, P. Turaga, and R. Chellappa, "Moving vistas: Exploiting motion for describing scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010.

[5] K. Derpanis, M. Lecce, K. Daniilidis, and R. P. Wildes, "Dynamic scene understanding: The role of orientation features in space and time in scene classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012.

[6] M. C. Potter and E. I. Levy, "Recognition memory for a rapid sequence of pictures." *Journal of experimental psychology*, vol. 81, no. 1, p. 10, 1969.

[7] G. A. Rousselet, S. J. Thorpe, M. Fabre-Thorpe *et al.*, "How parallel is visual processing in the ventral pathway?" *Trends in cognitive sciences*, vol. 8, no. 8, pp. 363–370, 2004.

[8] L. Fei-Fei, R. VanRullen, C. Koch, and P. Perona, "Rapid natural scene categorization in the near absence of attention," *Proceedings of the National Academy of Sciences*, vol. 99, no. 14, pp. 9596–9601, 2002.

[9] M. Marszalek, I. Laptev, and C. Schmid, "Actions in context," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009.

[10] K. Chatfield, V. Lempitsky, A. Vedaldi, and A. Zisserman, "The devil is in the details: an evaluation of recent feature encoding methods," in *Proc. Brit. Mach. Vis. Conf.*, 2011.

[11] J. Stone, *Vision and Brain - How we perceive the world*. MIT press, 2012.

[12] M. Szummer and R. Picard, "Indoor-outdoor image classification," in *Pro. IEEE Int. Workshop on Content-Based Access of Image and Video Database*, 1998.

[13] A. Oliva and A. Torralba, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *Int. J. Comput. Vis.*, vol. 42, pp. 145–175, 2001.

[14] A. Vailaya, M. A. T. Figueiredo, A. Jain, and H.-J. Zhang, "Image classification for content-based indexing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 10, no. 1, pp. 117–130, 2001.

[15] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005.

[16] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006.

[17] J. Liu and M. Shah, "Scene modeling using co-clustering," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2007.

[18] J. Vogel and B. Schiele, "Semantic modeling of natural scenes for content-based image retrieval," *Int. J. Comput. Vis.*, vol. 72, no. 3, pp. 133–157, 2007.

[19] N. Rasiwasia and N. Vasconcelos, "Scene classification with low-dimensional semantic spaces and weak supervision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008.

[20] N. M. Elfiky, J. González, and F. X. Roca, "Compact and adaptive spatial pyramids for scene recognition," *Image Vis. Comput.*, vol. 30, no. 8, pp. 492–500, 2012.

[21] M. Juneja, A. Vedaldi, C. V. Jawahar, and A. Zisserman, "Blocks that shout: Distinctive parts for scene classification,"

in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013.
[22] A. Chan and N. Vasconcelos, "Probabilistic kernels for the classification of auto-regressive visual processes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005.
[23] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Spacetime forests with complementary features for dynamic scene recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2013.
[24] ——, "Bags of spacetime energies for dynamic scene recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014.
[25] L. Wiskott and T. Sejnowski, "Slow feature analysis: Unsupervised learning of invariances," *Neural Computation*, vol. 14, no. 4, pp. 715–770, 2002.
[26] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Robust object recognition with cortex-like mechanisms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 3, pp. 411–426, 2007.
[27] C. Theriault, N. Thome, and M. Cord, "Dynamic scene classification: Learning motion descriptors with slow features analysis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013.
[28] K. Cannons and R. Wildes, "The applicability of spatiotemporal oriented energy features to region tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 784–796, 2014.
[29] K. Derpanis, M. Sizintsev, K. Cannons, and R. P. Wildes, "Action spotting and recognition based on a spatiotemporal orientation analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 3, pp. 527–540, 2012.
[30] A. Zaharescu and R. Wildes, "Anomalous behaviour detection using spatiotemporal oriented energies, subset inclusion histogram comparison and event-driven processing," in *Proc. Eur. Conf. Comput. Vis.*, 2010.
[31] M. Sizintsev and R. P. Wildes, "Spacetime stereo and 3D flow via binocular spatiotemporal orientation analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2241–2254, 2014.
[32] T. V. Papathomas, A. Gorea, and B. Julesz, "Two carriers for motion perception: Color and luminance," *Vision Research*, vol. 31, no. 11, pp. 1883–1892, 1991.
[33] S. Engel, X. Zhang, and B. Wandell, "Colour tuning in human visual cortex measured with functional magnetic resonance imaging," *Nature*, vol. 388, no. 6637, pp. 68–71, 1997.
[34] A. Gorea, T. V. Papathomas, and I. Kovacs, "Motion perception with spatiotemporally matched chromatic and achromatic information reveals a "slow" and a "fast" motion system," *Vision Research*, vol. 33, no. 17, pp. 2515 – 2534, 1993.
[35] B. Watson and A. Ahumada, "A look at motion in the frequency domain," in *Proceedings of the Motion Workshop*, 1983.
[36] G. Granlund and H. Knutsson, *Signal Processing for Computer Vision*. Kluwer Academic Publishers Norwell, MA, USA, 1995.
[37] L. Alvarez, F. Guichard, P. Lions, and J. Morel, "Axioms and fundamental equations of image processing," *Arch. Rat. Mech. Anal.*, vol. 123, no. 3, pp. 199–257, 1993.
[38] A. V. Oppenheim, R. W. Schafer, J. R. Buck *et al.*, *Discrete-time signal processing*. Prentice Hall, Upper Saddle River, New Jersey, USA, 1999, vol. 5.
[39] F. Perronnin, J. Sánchez, and T. Mensink, "Improving the Fisher kernel for large-scale image classification," in *Proc. Eur. Conf. Comput. Vis.*, 2010.
[40] Y. Boureau, F. Bach, Y. LeCun, and J. Ponce, "Learning mid-level features for recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010.
[41] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010.
[42] L. Cao, R. Ji, Y. Gao, Y. Yang, and Q. Tian, "Weakly supervised sparse coding with geometric consistency pooling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012.
[43] F. Perronnin and C. Dance, "Fisher kernels on visual vocabularies for image categorization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2007.
[44] X. Zhou, K. Yu, T. Zhang, and T. Huang, "Image classification using super-vector coding of local image descriptors," in *Proc. Eur. Conf. Comput. Vis.*, 2010.
[45] H. Jégou, F. Perronnin, M. Douze, J. Sánchez, P. Pérez, and C. Schmid, "Aggregating local image descriptors into compact codes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 9, pp. 1704–1716, 2012.
[46] Y. Boureau, N. Le Roux, F. Bach, J. Ponce, and Y. LeCun, "Ask the locals: multi-way local pooling for image recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011.
[47] J. Feng, B. Ni, Q. Tian, and S. Yan, "Geometric ℓ_p-norm feature pooling for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011.
[48] Y. Jia, C. Huang, and T. Darrell, "Beyond spatial pyramids: Receptive field learning for pooled image features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012.
[49] R. Wildes and J. Bergen, "Qualitative spatiotemporal analysis using an oriented energy representation," in *Proc. Eur. Conf. Comput. Vis.*, 2000.
[50] E. Adelson and J. Bergen, "Spatiotemporal energy models for the perception of motion," *Journal of the Optical Society of America*, vol. 2, no. 2, pp. 284–299, 1985.
[51] W. Freeman and E. Adelson, "The design and use of steerable filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 9, pp. 891–906, 1991.
[52] R. Arandjelovic and A. Zisserman, "Three things everyone should know to improve object retrieval," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012.
[53] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "C3D: Generic features for video analysis," *arXiv preprint arXiv:1412.0767*, 2014.
[54] C. Feichtenhofer, A. Pinz, and R. P. Wildes, "Dynamically encoded actions based on spacetime saliency," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2015.

Christoph Feichtenhofer received the BSc (with honors) and MSc degree (with honors) from the Graz University of Technology (TU Graz) in 2011 and 2013, respectively. He is a recipient of a DOC Fellowship of the Austrian Academy of Sciences at the Institute of Electrical Measurement and Measurement Signal Processing, TU Graz. His main areas of research include the development of spatiotemporal representations for dynamic scene understanding.

Axel Pinz (Member, IEEE) received the PhD degree from Vienna University of Technology in 1988 and the habilitation from Graz University of Technology, Austria in 1995. In 1983, he joined the Institute for Surveillance and Remote Sensing at the University of Natural Resources, Vienna, where he developed vision algorithms for remote sensing. From 1990-1994, he was with the Department for Pattern Recognition and Image Processing, Vienna University of Technology. Since 1994, he has been with Graz University of Technology, where he is heading a research group on vision-based and optical measurement. His main research interests are in object and video categorization.

Richard P. Wildes (Member, IEEE) received the PhD degree from the Massachusetts Institute of Technology in 1989. Subsequently, he joined Sarnoff Corporation in Princeton, New Jersey, as a Member of the Technical Staff in the Vision Technologies Lab. In 2001, he joined the Department of Electrical Engineering and Computer Science at York University, Toronto, where he is an Associate Professor and a member of the Centre for Vision Research. Honours include receiving a Sarnoff Corporation Technical Achievement Award, the IEEE D.G. Fink Prize Paper Award for his Proceedings of the IEEE publication Iris recognition: An emerging biometric technology and twice giving invited presentations to the US National Academy of Sciences. His main areas of research interest are computational vision, as well as allied aspects of image processing, robotics and artificial intelligence.