Dynamic Texture Recognition Based on Compression Artifacts

Dubravko Ćulibrk[1], Matei Mancas[2], and Vladimir Ćrnojevic[1]

Abstract The paper proposes a novel approach to the classification of compressed videos containing dynamic textures. The term dynamic texture is usually used with reference to image sequences of various natural processes that exhibit stochastic dynamics (e.g., water, fire and windblown vegetation). Description and recognition of dynamic textures have attracted growing attention.

Although one of the most important prospective applications of the technology is content-based video retrieval, recognition of dynamic textures for compressed video has not been considered. The content of video and dynamic textures in particular, profoundly affect the performance of video compression algorithms. The prominence of compression artifacts can, therefore, be used to recognize dynamic textures in compressed videos. In the paper, we show how features, previously proposed for quality assessment, statistical analysis and a soft computing technique (neural networks) can be used to discern 23 different classes of dynamic textures in a standard video database, with 99.5% accuracy.

1 Introduction

Dynamic textures represent a set of phenomena occurring in nature, where the perceived changes in the appearance of a system of large number elements are consistent, although the individual elements undergo stochastic changes in theirs. Typically the changes are due to motion (e.g. turbulent water water, smoke, vegetation in the wind, insect swarms), but may be the result of the changing intensity of light emitted (e.g. fire). In the computer vision literature, such patterns have appeared collectively under various names,

[1] University of Novi Sad, Serbia, {dculibrk,crnojevic}@uns.ac.rs
[2] University of Mons, Belgium, matei.mancas@umons.ac.be

including, turbulent flow/motion, temporal textures, time-varying textures, dynamic textures, and textured motion [6]; the term dynamic texture will be used herein. Zhao and Pietikinen consider such phenomena extensions of the static texture to the temporal domain [26], since the effect is that of a textured object undergoing transformations. Derpanis and Wildes [6], however, point out that the term can apply equally well to simpler phenomena when analyzed in terms of aggregate regional properties (e.g., orderly pedestrian crowds and vehicular traffic).

The ability to recognize dynamic textures based on visual processing is of significance to a number of applications, including, video indexing/retrieval, surveillance and environmental monitoring where they can serve as keys, isolate background clutter (e.g., fluttering vegetation) from activities of interest and detect various critical conditions (e.g., fires), respectively. It comes as no surprise that a significant amount of research effort has been directed toward solving this problem [4] [26] [12] [6]. However, to the best of our knowledge, no one has dealt with the possibility of recognizing dynamic textures in compressed (coded) videos, although this is the 'natural' state of the material in applications such as content-based video retrieval and the preferred way of storing and transmitting visual data in all other.

The quality of coded video sequences depends on the video codec, bit-rates required and the content of video material [5]. Clearly, if the bit-rate and the codec are the same over a range of sequences - a reasonable assumption for multimedia databases - the quality of compressed videos is dependent only on the content. We propose a video classification approach that exploits this relationship. Using the compressed videos available in a standard database used for dynamic texture recognition [12], we show that the measures of the level of artifacts introduced by the coding algorithm can be used as basis for efficient dynamic texture recognition.

Based on video quality measures, content-dependent features are extracted for the frames of the video. These are then aggregated so that each video sequence is represented by a fixed-length signature derived from the feature-values obtained for single frames. Video Signatures (VS) are subsequently used to train a boosted soft computing (neural network) classifier [2]. Experimental results obtained through cross-validation show that the classifier is able to achieve perfect (99.5% accurate) classification for the data set used.

The paper makes several contributions. Dynamic texture classification from compressed video is considered for the first time. To the best of our knowledge, no one has attempted to use the correlation between coding artifacts, video quality and content to classify dynamic textures. The proposed methodology relies on a state-of-the-art Video Quality Assessment (VQA) approach and exploits visual saliency due to motion to extract dynamic-texture related changes. Finally we propose the use of boosting Multi Layer Perceptron (MLP) neural networks to classify dynamic textures - another novelty- and show that such a classifier, combined with the proposed features, can

(a) water (b) tentacle (c) smoke

(d) vegetative (e) fire (f) insects

Fig. 1 Sample frames from DynTex sequences

achieve 99.5% accurate classification for 23 classes of dynamic textures represented in a standard video data set.

The rest of the paper is organized as follows: Section 2 deals with the relevant published work. Section 3 describes the methodology proposed. Section 4 discusses experiments performed and results achieved. Section 5 is dedicated to our conclusions.

2 Related Work

2.1 Dynamic Texture Classification

The research into the classification and recognition of dynamic textures continues unabated [6][26]. A large number of approaches have been proposed over the last ten years. In their 2005 survey Chetverikov and Péteri [4] divided the existing approaches into five classes: methods based on optic flow, methods computing geometric properties in the spatiotemporal domain, methods based on local spatiotemporal filtering, methods using global spatiotemporal transforms and model-based methods that use estimated model parameters as features. Regardless of the type of the approach, they attempt to extract features descriptive of the dynamic texture and classify them by either defining a suitable distance measure and creating a simple distance-based algorithm for comparison or training a machine learning algorithm to achieve the task.

In their 2007 paper [26], Zhao and Pietikinen proposed volume local binary patterns (VLBP) as features to describe dynamic textures. The VLBPs are an extension of the LBP operator widely used in ordinary texture analysis, that combine motion and appearance. They tested their approach using videos

generated by extracting parts of the sequences in the DynTex database [12], creating a data set that had 10 examples of a certain class derived from single DynTex sequences. Their classifier is a simple nearest neighbor classifier, based on the log-likelihood statistic that allows them to compare VLBPs, and they used leave-one-group-out (i.e. n/m fold cross-validation [22]) to measure performance, where m corresponds to the number of examples extracted for a single dynamic texture and n is the total number of examples. Various classification rates were achieved depending of whether or not the features used were shift-invariant and how long the feature vector was. Their best result is an accurate classification rate of 95.71%, achieved for a shift-invariant VLBP and a fairly large feature vector ($4,176$ bins) .

Chan and Vasconcelos [3] model the dynamic texture as a linear dynamic system (LDS) and achieve good classification using Martin distance to compare the models. They evaluated both nearest neighbor and support vector machine (SVM) classifiers and showed that the use of a machine learning algorithm such as SVM can improve the classification significantly. Through the use of the SVM classifier they achieved accurate classification rate of 97.5% on the UCLA database [15]. More recently (2009) their work has been extended by Ravichandran *et al.* [13] to use bags of LDSs to achieve improved view-invariant texture classification, when eight classes of textures are concerned.

Recently (2010), Derpanis and Wildes [6] proposed new features based on spatiotemporal oriented energy filters to describe dynamic textures and classify them. They identified 7 semantic categories in the UCLA database (flames, fountain, smoke, turbulence, waves, waterfall, vegetation) and achieved a comparatively low classification rate of 92.3%, on sequences derived from this database. However, they specifically considered shift-invariant recognition, and report improved performance under this conditions.

To the best of our knowledge no one has considered the problem of classifying dynamic texture in compressed videos nor the use of features used to measure different artifacts introduced by coding, as basis for dynamic texture classification/recognition.

2.2 Video Quality Assessment

The quality of coded video sequences depends on the video codec, bit-rates required and the content of video material [5]. If the bit-rate and the codec are the same over the range of sequences the quality of compressed videos is dependent only on the content, allowing for the use of features related to quality to discern content. This is not an unrealistic constraint, e.g. a quick calculation reveals that environ 290 million videos have been uploaded to YouTube in 2010 [24], all using the same codec and a significant subset with default parameter settings.

Overall degradation in the quality of the sequence, due to encoder/decoder implementations as part of transport stream at various bit rates, is a compound effect of different coding artifacts [21]. Three types of artifacts are typically considered pertinent to block coded data: blocking, ringing and blurring. Blocking appears in all block-based compression techniques, which include all contemporary codecs [14] [8], due to coarse quantization of frequency components [19] [20]. It can be observed as surface discontinuity (edge) at block boundaries. These edges are perceived as abnormal high frequency components in the spectrum. Blocking is usually masked by the presence of strong texture in the background and blockiness measures are designed to estimate what part of the discontinuity on the block edges is due to the blocking effect vs. the texture in the content [1]. In the setup proposed in this paper, blockiness measures are related to the texture in the background. Ringing is observed as periodic pseudo edges around original edges [11]. It is due to improper truncation of high frequency components. In the worst case, the edges can be shifted far away from the original edge locations, observed as false edge. Blurring, which appears as edge smoothness or texture blur, is due to the loss of high frequency components when compared with the original image. Blurring causes the received image to be smoother than the original one [7] and the measures of blurring try to estimate the difference in activity of the original content with respect to coded version. They are profoundly influenced by the textures in the video content.

A large number of published papers exist that propose different measures of prominent artifacts which appear in coded images and video sequences [19] [5]. In this study we limit ourselves to no-reference approaches, where only compressed video is available. This is a harder problem, but more realistic in applications such as video-retrieval.

Several published approaches to measuring video quality are of interest for the discussion in the following sections. Wang *et al.*[19] proposed a no-reference approach to quality assessment in JPEG coded images. Their final measure is derived as a non-linear combination of a blockiness, local activity and a so-called zero-crossing measure. The combination is supposed to provide information regarding both blockiness and blurring (via the two latter measures) in JPEG coded images. More recently, Babu *et al.* [1] proposed a blockiness measure for use in VQA, which takes effects along each edge of the block into account separately.

Measures related to various artifacts are usually evaluated for each frame of the sequence and collapsed temporally to arrive at a quality measure for the whole sequence [23] [18] [10] [5].

Recently, Culibrk *et al.* [5] proposed a VQA approach that improves quality estimation by separately considering the regions of the frame in which salient-motion is present and the rest of the frame. Using a simple multi-scale foreground-background segmentation approach, they detect the salient regions and calculate a number of features related to the observed temporal changes. In addition, they calculate the blockiness and blurring measures pro-

posed by Babu *et al.* [1] and Wang *et al.* [19] for the salient and non-salient parts of the frame. Using these features they train a neural-network and a decision tree classifier that are able to achieve state-of-the art quality estimation on a per-frame basis. The final estimate of the quality is the median value obtained for the frames of the sequence.

The approach of Culibrk *et al.* has been selected as a state-of-the art approach for measuring video quality that is used in the study presented here. Since dynamic textures are by their very nature salient due to motion, this approach enables us to capture the features related to the dynamic-texture part of the sequence frames and filter out the rest of the sequence.

2.3 Classification

Once descriptive features are extracted the preferred approaches to classification of dynamic textures seem to be the Nearest Neighbor (NN) classifier and Support Vector Machines (SVM) [26] [3] [13] [6].

Culibrk *et al.* [5], in the other hand, proposed using either a decision tree classifier or a Multi Layer Perceptron (MLP) [9] neural network to estimate the quality of video based on their features. In addition, they performed automatic feature selection to evaluate the impact of saliency and showed that an MLP estimator can achieve good results using a subset of just 5 features.

Here we propose using an MLP based classifier to discern different texture classes. Neural networks represent a class of machine learning algorithms designed to follow the basic principles of biological neural cells and as such fall into the domain of soft computing. They consist of a number of interconnected nodes that receive signals through their input connections, do simple processing and pass the output to other neurons. The connections between the neurons emulate the synapses between the neurons in biological systems and are assigned weights that code the relative influence between the connected nodes. In artificial neural networks the weights are learned from data in order to create classifiers or estimators with the desired behavior.

To enhance the performance of the MLP classifier we propose using a meta-learning algorithm Adaptive Boosting [16]. The effect of such an approach is analogous to creating a cascade of neural network classifiers, each trained on the set of examples that are incorrectly classified by the preceding stages. Schwenk and Bengio [17] discuss the merits of such an approach in detail.

3 Proposed Approach

A block diagram of the proposed dynamic texture classification approach is shown in Fig 2. The input data are compressed videos of dynamic textures. If the video is not compressed it can easily be coded with any lossy compression algorithm. Each video in the data set is initially processed to extract measures related to motion, salient changes, blurring and blockiness. A total of 17 measures is extracted for half of the frames of video, distributed uniformly - once the measures have been calculated for a frame, the next frame is skipped. This increases the efficiency and has no impact on the effectiveness. The values of measures for all frames of a single video are clustered into 10 clusters using k-means clustering [25]. The process yields 10 cluster centroids that represent each video. The set of centroids is a fixed-size representation of a video, regardless of the number of frames it has. This is referred to as a *Video Signature* (VS).

Once the signatures for all the videos are computed they are used to train and test a neural-network-based meta classifier. Each centroid is used as a separate input the classifier and each video is represented by 10 centroids comprising the signature. This is done to make the approach less sensitive to the measurement error introduced by the saliency-detection module, which takes approximately 50 frames to adapt to sudden scene changes and to learn the background model when presented with a new sequence.

The classifier is used to discern the semantic class for each of the 10 centroids comprising a VS. The mode (most common value) of the 10 class labels obtained in this manner is used the final classification of the video.

3.1 Extracting Features

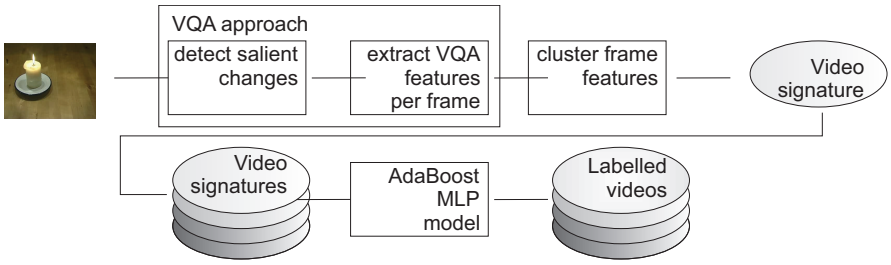

Fig. 2 Proposed video description approach: A video signature is extracted for each video, the signatures used to train a boosted MLP classifier that can be used to classify other videos

3.2 Basic Features for Video Classification

Set of features related to video quality was adopted from the work of Culibrk *et al.* [5].

The approach proposed in [5] attempts to estimate salient motion in each frame of the sequence by performing background subtraction at several different scales. The scaled frames are obtained from a frame of the sequence by performing spatial Gaussian filtering and decimating the frame to get the next scale. This yields a representation of each frame of the sequence in the form of a Gaussian pyramid. The same process is applied to background frames. The results of background subtraction at each scale are thresholded to eliminate small changes and summed up to form a single saliency map. Outlier detection is then used to determine which parts of the map are salient and which are not. Even with a small number of scales (3-5), the approach is able the achieve meaningful, if somewhat coarse, segmentation of interesting moving objects in the scene. In the case of the DynTex database, this corresponds to the dynamic-texture regions of the frame. The process is illustrated in Figure 3.

Once the salient parts of the frame have been determined several basic features are used to describe the salient motion in a frame: number of salient regions, their average size, and first moments (mean and standard deviation) of the difference between the current frame and background frames, calculated separately for salient and non-salient regions.

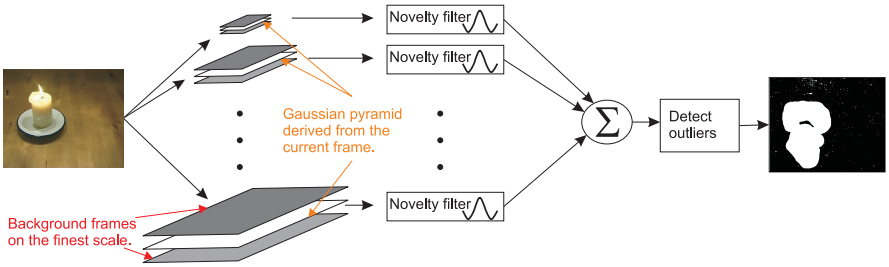

Fig. 3 Salient-motion region segmentation.

Also, to account for blurring and blockiness, Z-score measures proposed by Wang *et al.* [19] and the blockiness measure proposed by Babu *et al.* [1] are calculated separately for salient, non-salient and (in the case of the last feature) border regions. This should provide a good description of the texture within the different regions of the frame.

The blockiness measures proposed by Wang *et al.* and Babu *et al.* are profoundly different. Babu *et al.* focus on the effects that can be observed along the edges of a single block. Their measure is designed to detect blocks

with low spatial activity along the edges, but significant differences across them.

To characterize the activity on the inside of the block edge they calculate the standard deviation of pixel values for 6-pixel long stretches along he border of the block, since they observed that blockiness that spans less than 6 pixels is not perceived as significant. For each edge of the block they try to detect if there is significant activity that could mask the blockiness effect. Let $\{I_{k,j}|k \in [1,4], j \in [1,8]\}$ be the edges of a block and $\{O_{k,j}, k \in [1,4], j \in [1,8]\}$ the corresponding pixels across the edge of the block. We first consider the standard deviation of pixel values on the inside of block edges:

$$\sigma_{k,j} = stddev(I_{k,j}), k \in [1,3], j \in [k, k+5] \tag{1}$$

Then we compute the gradient across block edges for each subsegment of the edge:

$$\Delta_{k,j} = mean(|I_{k,j} - O_{k,j}|), k \in [1,3], j \in [k, k+5] \tag{2}$$

If any of $\sigma_{k,i}$ is below an empirically selected threshold ε, than that edge can contribute to the blockiness, but it will do so only if the gradient is larger than a different threshold τ. For a block i of a frame, we define

$$W_i = \begin{cases} 1, (\exists \sigma_{k,j}, \Delta_{k,j})(\sigma_{k,j} < \varepsilon \wedge \Delta_{k,j} > \tau) \\ 0, \text{otherwise.} \end{cases} \tag{3}$$

Finally, we calculate the proportion of blocks that contributes to the blockiness effect as the measure of blockiness:

$$BB = \frac{\sum_{i=1}^{NB} W_i}{\mathrm{NB}} \tag{4}$$

where NB is the number of blocks in the region considered.

The authors of the approach [1] suggest $\varepsilon = 0.1$ and $\tau = 2.0$, which are also the values used in the study presented here.

The approach of Wang *et al.* is based on the observation that the artifacts can be detected if the image is transformed to the frequency domain and its power spectrum examined. They design their measures of blurring and blockiness in an attempt to achieve a less computationally intensive approach than that of computing the full power spectrum. Let $x(m,n)$ $m \in [1,M]$ and $n \in [1,N]$, be the pixel values (signal) for a frame. First a differencing signal is calculated along the horizontal lines:

$$d_h(m,n) = x(m,n+1) - x(m,n), n \in [1, N-1] \tag{5}$$

The blockiness measure proposed by Wang *et al.* tries to take into account the differences between a whole line of blocks, rather than looking at a single block:

$$B_h = \frac{1}{M(\lfloor N/8 \rfloor - 1)} \sum_{i=1}^{M} \sum_{j=1}^{\lfloor N/8 \rfloor - 1} d_h(i, 8j) \tag{6}$$

Thus, the Wang *et al.* provides a more wider-range measure of blockiness, when compared to the basic Babu *et al.* metric.

Wang *et al.* proposed two measures in an attempt to characterize the spatial activity of the signal. Their motivation lies in the fact that activity is reduced by blurring. The activity is related to how pronounced the texture is in a particular region of the frame. The first measure is the average absolute difference between in-block image samples:

$$A_h = \frac{1}{7} \left[\frac{8}{M(N-1)} \sum_{i=1}^{M} \sum_{j=1}^{N-1} |d_h(i,j) - B_h| \right] \tag{7}$$

The second measure is the zero-crossing (ZC) rate. They define for $n \in [1, N-2]$:

$$z_h(m,n) = \begin{cases} 1, \text{ horizontal ZC at } d_h(m,n) \\ 0, \text{ otherwise} \end{cases} \tag{8}$$

the horizontal ZC rate can then be estimated as:

$$Z_h = \frac{1}{M(N-2)} \sum_{i=1}^{M} \sum_{j=1}^{N-2} z_h(m,n) \tag{9}$$

The vertical features (B_v,A_v and Z_v) are then calculated in a similar fashion. The overall blockiness, activity and ZC rate are calculated as:

$$B = \frac{B_h + B_v}{2}, A = \frac{A_h + A_v}{2}, Z = \frac{Z_h + Z_v}{2} \tag{10}$$

Finally the formulate an empirical model for the quality score:

$$Z_{score} = \alpha + \beta B^{\gamma_1} A^{\gamma_2} Z^{\gamma_3} \tag{11}$$

They used the non-linear regression routine available in the Matlab statistics toolbox to find the best value of parameters (α,β,γ_1,γ_2,γ_3) for Eq. 11. The values they calculated are used in the study presented here: $\alpha = -245.9$, $\beta = 261.9$, $\gamma_1 = -0.0024$, $\gamma_2 = 0.016$, $\gamma_3 = 0.0064$.

Blockiness is masked by the texture(spatial activity) in the region for which it is calculated. Activity measures are directly related to texture properties within blocks. The final Z-score is a nonlinear combination of these measures, that emulates the properties of the human visual system.

All the quality related features used are listed in Table 1. In the scope of the study presented here, they are good features to describe two regions of interest in test videos. Dynamic texture, which forms the salient part of the frame and the background which is non-salient.

It should be noted that both Wang *et al.* and Babu *et al.* measures were originally designed for 8×8 block size, which is the only size available in MPEG-2 [8], but not in MPEG-4/H.264/AVC [14]. However, since the size of the blocks in the latter case is constrained to 16×16, 8×8 or 4×4, the measures should be able detect blockiness and blurring along a subset of edges and within part of the blocks, and therefore can be used for any block-based codec.

Table 1 List of used quality features.

Salient reg. count	Z_{score} non-salient
Avg. reg. size	A salient
Mean change non-salient	B salient
Change Std.Dev. non-salient	Z salient
Mean Change salient	Z_{score} salient
Change Std.Dev. salient	BB non-salient
A non-salient	BB salient
B non-salient	BB border
Z non-salient	

4 Experiments and Results

4.1 Data Set

Videos from the DynTex data set [12] were used to evaluate the approach. The DynTex data set contains more than 650 varied dynamic texture videos, but the information about the type of textures shown in the sequences is not provided for all the videos in the set. Figure 1 shows example textures from this data set. The image size is 352×288 and the compressed videos provided were coded using DivX codec, i.e. an MPEG-4 Part 2 codec.

A subset of 202 sequences, spanning some 23 classes of very varied dynamic textures has been selected to evaluate the proposed approach. The subset is comprised of those DynTex sequences that were labelled as containing a single class of dynamic texture including those labelled NA for which the class information is not available. We treat the NA sequences as an additional class, increasing the diversity of the test set. The texture classes contained in our data set are: textile, vegetation, grass, NA, streaks, water, steam, fire, smoke, branch, cloud, leaf, car, flower, needle, fur, fish, tentacle, insects, CD, foam, light and paper. Sample frames from some of the sequences are shown in Figure 1.

4.2 Classification

Once the video signatures have been extracted, we used the Wakaito Environment for Knowledge Discovery (WEKA) tool [22] to train and test our classifiers. WEKA is an open source data mining and machine learning environment, which allows for different machine learning algorithms to be tested on a data set. The evaluation procedure conducted in two phases.

In the first phase we experimented with various algorithms, including the MLP suggested by Culibrk *et al.*, various decision trees and AdaBoost based on different algorithms. To determine the most suitable classifier, they were tested using the 10 fold stratified cross-validation methodology, i.e. 10% of data was withheld during training and used for testing. The data was selected randomly, but care was taken to preserve the distribution of the classes that exists in the original data set, since not all of the classes were represented with the same number of sequences.

In the second phase the best algorithm identified during the first stage was tested by withholding the VS of a single sequence for testing. The rest of the data was used for training. Once the classification for each part of the test VS was done, the mode of the VS was used to assign a class label to the sequence itself.

In our experiments, AdaBoost-ing the MLP classifier achieved the best result. In the first phase of the experiments, the classifier achieved 96.4% correct classification of each video signature part when tested on the training set and a 87.87% accuracy when cross-validated. The confusion matrix obtained for VS-part classification is shown in Figure 4 In our experiment we used AdaBoost M1 method, with 10 iterations. The MLPs contained 100 neurons, were trained using back-propagation with the learning rate of 0.3, moment 0.2 and 500 epochs.

In the second phase, since we used the mode of 10 classification values obtained per VS as the final label of the video, the cross-validated performance was nearly perfect (99.5% accurate) at the video-sequence level. The classifier failed to classify a single sequence accurately, out of 202 sequences in the data set. It should be noted that the sequence that the classifier failed to classify was one of 15 sequences in the database that are missing class information and is therefore quite possible that the training set contained no information that would enable the classifier to learn that particular case.

5 Conclusion

Although content-based video retrieval and indexing are usually stated as important potential applications of dynamic texture classification and recognition methodologies, there has so far been no attempt to address the problem of classifying dynamic textures based on compression artifacts.

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	← classified as
195	0	0	3	0	13	1	0	1	0	3	0	0	0	0	0	0	6	0	1	1	6	0	a = textile
1	98	1	0	0	2	0	0	1	0	0	1	0	0	1	0	0	4	0	0	0	1	0	b = vegetation
0	0	53	0	0	6	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	c = grass
4	1	0	128	0	6	0	0	0	1	1	0	1	0	0	0	0	7	0	0	0	1	0	d = ?
0	0	0	0	19	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	e = streaks
6	2	5	2	0	583	1	0	2	1	6	0	2	1	0	0	6	5	0	0	7	1	0	f = water
2	0	0	0	0	1	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	g = steam
6	0	0	0	0	1	1	10	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	h = fire
2	0	0	0	0	2	0	0	31	0	1	1	0	1	0	0	2	0	0	0	0	0	0	i = smoke
1	0	0	1	0	1	0	0	0	97	0	0	0	0	0	0	0	0	0	0	0	0	0	j = branch
0	0	0	0	0	5	1	0	0	0	54	0	0	0	0	0	0	0	0	0	0	0	0	k = cloud
0	0	0	0	0	1	0	0	0	0	0	8	0	0	0	0	0	1	0	0	0	0	0	l = leaf
1	0	0	1	0	2	0	0	2	0	0	0	44	0	0	0	0	0	0	0	0	0	0	m = car
2	1	0	0	0	7	0	1	0	1	0	0	0	33	0	0	0	4	0	0	0	0	1	n = flower
0	0	0	0	0	2	0	0	0	0	0	0	0	0	18	0	0	0	0	0	0	0	0	o = needle
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	8	0	1	0	0	0	0	0	p = fur
0	3	0	1	0	3	0	0	1	0	0	0	0	1	0	0	35	6	0	0	0	0	0	q = fish
4	3	3	0	1	7	0	0	0	1	0	0	0	0	0	1	1	149	0	0	0	0	0	r = tentacle
0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	1	0	7	0	0	0	0	s = insects
0	0	0	0	0	2	0	0	0	0	0	0	1	0	0	0	0	0	0	27	0	0	0	t = CD
0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	u = foam
3	1	0	1	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	102	0	v = light
1	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	15	w = paper

Fig. 4 Confusion matrix for the proposed classifier.

Since the performance of coding algorithms in terms of resulting video quality is profoundly dependent on the content of video, a novel approach to dynamic texture classification, which exploits this link is proposed in the paper. Specifically we showed that features commonly used for video quality assessment can be used, efficiently, to discern between different dynamic textures. The assumption made is that the videos are coded using the same codec and same bit-rates, which is not unreasonable in case of large multimedia databases.

An MLP-based AdaBoost classifier has been trained and evaluated using video quality features obtained through a state-of-the-art video quality assessment approach. A standard set of compressed dynamic texture videos has been used to test the approach. The approach achieves nearly perfect classification (99.5%) when cross-validated.

Several venues should be explored for further studies. The approach should be tested using other available databases, such as the UCLA database [15]. More importantly, the approach should be tested for different codecs. The blockiness and blurring measures designed may have to be adapted to handle the variable block-size of state-of-the art codecs explicitly.

Acknowledgements This research has in part been financed by the COST action IC0702 (SoftStat).

References

1. Babu V, Andrew P, Inge HO (2008) Evaluation and monitoring of video quality for UMA enabled video streaming systems. *Multimedia Tools Appl.* 37(2):211–231
2. Breiman L (2001) Random forests. *Machine Learning* 45(1):5–32

3. Chan A, Vasconcelos N (2007) Classifying video with kernel dynamic textures. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1–6. IEEE Press, Piscataway
4. Chetverikov D, Péteri R (2005) A brief survey of dynamic texture description and recognition. *Computer Recognition Systems* 17–26
5. Culibrk D, Mirkovic M, Zlokolica V, Pokric M, Crnojevic V, Kukolj D (2010) Salient Motion Features for Video Quality Assessment. *IEEE Trans. on Image Processing* 948–958
6. Derpanis K, Wildes R (2010) Dynamic texture recognition based on distributions of spacetime oriented structure. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 191–198. IEEE Press, Piscataway
7. Ferzli R, Karam L (2007) A no-reference objective image sharpness metric based on just-noticeable blur and probability summation. *Proc. IEEE Conf. on Image Processing (ICIP)*, III:445–448. IEEE Press, Piscataway
8. Haskell B, Puri A, Netravali A (1997) *Digital video: an introduction to MPEG-2*. Kluwer, Amsterdam, Netherlands
9. Haykin S (1994) *Neural Networks: A Comprehensive Foundation*. Macmillan, New York
10. Kim K, Davis L (2004) A fine-structure image/video quality measure using local statistics. *Proc. IEEE Conf. on Image Processing* V:3535–3538
11. Kirenko I (2006) Reduction of coding artifacts using chrominance and luminance spatial analysis. *Proc. Int. Conf. on Consumer Electronics (ICCE)*, 209–210
12. Péteri R, Fazekas S, Huiskes M (2010) DynTex: A comprehensive database of Dynamic Textures. *Pattern Recognition Letters*
13. Ravichandran A, Chaudhry R, Vidal R (2009) View-invariant dynamic texture recognition using a bag of dynamical systems. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 1651–1657. IEEE Press, Piscataway
14. Richardson I (2003) *H.264 and MPEG-4 video compression*. Wiley Online Library
15. Saisan P, Doretto G, Wu Y, Soatto S (2001) *Dynamic texture recognition*
16. Schapire R (2003) The boosting approach to machine learning: An overview. *Nonlinear Estimation and Classification*, 149–172 Springer, New York
17. Schwenk H, Bengio Y (2000) Boosting neural networks. *Neural Computation* 12(8):1869–1887
18. Wang Z, Lu L, Bovik A (2004) Video quality assessment based on structural distortion measurement. *Signal processing: Image communication* 19(2):121–132
19. Wang Z, Sheikh HR, Bovik AC (2002) No-reference perceptual quality assessment of jpeg compressed images. *Proc. IEEE Int. Conf. on Image Processing*, 477–480
20. Warwick G, Thong N (2004) *Signal Processing for Telecommunications and Multimedia*, Chapter 6: Classification of Video Sequences in MPEG Domain. Springer, New York
21. Winkler S (2005) *Digital video quality: vision models and metrics*. J. Wiley & Sons, Chichester, United Kingdom
22. Witten IH, Frank E (2005) *Data Mining: Practical machine learning tools and techniques, 2nd Edition*. Morgan Kaufmann, San Francisco
23. Wolf S, Pinson M (1999) Spatial-temporal distortion metrics for in-service quality monitoring of any digital video system. *Proc. Int. Symp. on Voice, Video, and Data Communications*. SPIE, Boston
24. YouTube: http://www.youtube.com/
25. Zechner M, Granitzer M (2009) Accelerating k-means on the graphics processor via CUDA. *Proc. Int. Conf. on Intensive Applications and Services (INTENSIVE 2009)*. IEEE Press, Piscataway
26. Zhao G, Pietikainen M (2007) Dynamic texture recognition using local binary patterns with an application to facial expressions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 915–928