Authorship Confirmation

As corresponding author I, <u>Mithilesh Kumar JHA</u>, hereby confirm on behalf of all authors that:

1. This manuscript, or a large part of it, <u>has not been published, was not, and is not being submitted to</u> any other journal.

2. If <u>presented at</u> or <u>submitted</u> to or <u>published at</u> a conference(s), the conference(s) is (are) identified and substantial <u>justification for re-publication</u> is presented below. A <u>copy of conference paper(s) is(are)</u> uploaded with the manuscript.

3. If the manuscript appears as a preprint anywhere on the web, e.g. arXiv, etc., it is identified below. <u>The preprint should include a statement that the paper is under consideration at Pattern Recognition Letters.</u>

4. All text and graphics, except for those marked with sources, are <u>original works</u> of the authors, and all necessary permissions for publication were secured prior to submission of <u>the manuscript.</u>

5. All authors each made a significant contribution to the research reported and have <u>read</u> and <u>approved</u> the submitted manuscript.

Signature <u>Mithilesh Kumar JHA</u>   Date <u>27-11-2013</u>

List any pre-prints: Not Applicable

Relevant Conference publication(s) (submitted, accepted, or published):   Video Compression Scheme Using DEMD Based Texture Synthesis in Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), 2011

Justification for re-publication:
The current submission differs considerably from the above preliminary work in the following ways. The earlier work encoded the IMFs' maxima and minima for multiple IMF levels, instead of the more efficient residue encoding in the current work. Moreover, the temporal information encoding in the H.264/MPEG framework - is a major contribution of the current submission.

ELSEVIER

DEMD-based Video Coding for Textured Videos in an H.264/MPEG Framework

Mithilesh Kumar Jha[a,**], Sumantra Dutta Roy[a], Brejesh Lall[a]

[a]*Department of Electrical Engineering, Indian Institute of Technology Delhi, Hauz Khas, New Delhi - 110 016, INDIA.*

ARTICLE INFO

Article history:

Video Coding
Texture Synthesis
Directional Empirical Mode Decomposition (DEMD)
Intrinsic Mode Functions(IMF)

ABSTRACT

This paper proposes a DEMD-based textured video coding scheme for efficient representation of textured data compatible with a H.264/MPEG framework. We encode the DEMD residue image of all non-*I* frames (*P* and *B* frames in H.264/MPEG) instead of the original frame. All non-*I* frames are synthesised using correlation search and patch-based multi-level IMF synthesis from the DEMD representation of key frames (*I* frames, in the H.264/MPEG framework) and the DEMD residual image of the frame being synthesised. Experimental results show that the proposed scheme provides better compression results as compared to MPEG/H.264 with acceptable reconstruction quality.

1. Introduction

Textures are homogeneous patterns that contain spatial, temporal, statistical and perceptual redundancies, which intensity or colour alone cannot describe adequately. Video applications (such as representing textured backgrounds in video conferencing, or tele-teaching scenarios) demand high compression with good perceptual reconstruction quality, instead of bit-accurate (high PSNR) reconstruction. This is because the human brain is able to decipher important variations in data at scales smaller than those of the viewed object.

Conventional video compression schemes such as MPEG2 [1] and H.264 [2] exploit statistical dependencies on the entire signal subject to a mean squared error (MSE) criterion. Efficient exploitation of statistical dependencies does not always correspond to the best psycho-visual result. Representing textured video data using conventional approaches (MPEG2/H.264, for instance) is not efficient primarily because of three reasons, (a) they are based on Shannon-Nyquist sampling [3], thus do not exploit inherent sparsity present in the signal, (b) acquisition and encoding are carried out on the entire signal, while most of the transformed data is discarded in the compression processes and thereby significantly wasting storage resources and

high computational cost (This is more relevant in case of texture video because of the fine detail and high frequency content), (c) these schemes do not exploit the perceptual redundancy present in the video data, which could result in much higher compression at an acceptable psycho-visual quality. The goal of a novel contemporary lossy compression scheme should be to reduce the entropy while preserving the perceptual quality of the frame using high-level computer vision tools and techniques. Most of these high-level processing techniques results in a small number of semantically relevant features, which can represent the salient features of the entire signal very accurately. Given this paradigm, a number of technical challenges arise, (i) how to identify an algorithm that can pre-process the video signal and how to identify the segment that should be represented by MSE-accurate criterion, (ii) how to create an approximation with subjective correspondence, (iii) how to represent a picture that consists of a mix of two approaches to compression.

In this paper, we propose a DEMD-based textured video coding scheme for efficient representation of textured data. We encode the DEMD residue image of all non-*I* frames (which due to the representation, are expected to have a significantly lower entropy, and can therefore be represented using fewer bits (Fig. 1 shows an example of the compression gain due to the DEMD-based entropy reduction for sample textured videos used in this paper) of all non-*I* frames (*P* and *B* frames in H.264/MPEG standard) in a H.264/MPEG framework. All non-*I* frames are synthesised using correlation search and patch-

---

[**]Corresponding Author

 e-mail: jham73@gmail.com (Mithilesh Kumar Jha),
sumantra@ee.iitd.ac.in (Sumantra Dutta Roy),
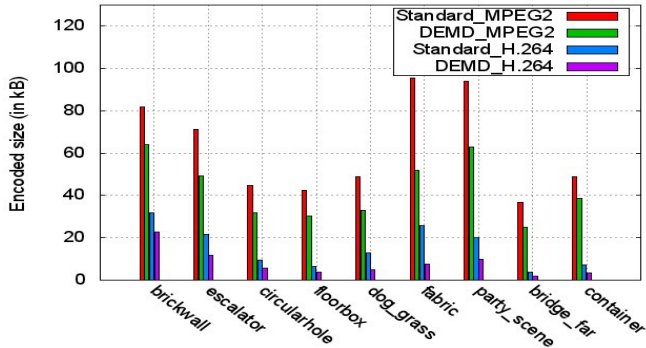brejesh@ee.iitd.ac.in (Brejesh Lall)

Fig. 1. DEMD residue encoding of *I* frames usually leads to better compression as compared to encoding the original *I* frame: results for some standard test sequences.'Standard_MPEG2/H.264' indicates *I* frame encoding, and 'DEMD-MPEG2/H.264' indicates DEMD residue encoding in an MPEG2/H.264 framework.

based multi-level IMF synthesis from the DEMD representation of key frames (*I* frames, in the H.264/MPEG framework) and the DEMD residual image of the frame being synthesised.

Existing methods of textured video compression can be broadly classified as parametric or non-parametric. Parametric approaches can achieve very high compression at low computational cost. Structural textures (with non-stationary data content) primarily have a large number of similar ordered patterns (texture subunits). Parametric models lose their compression gain advantage as the number of parameters increases considerably. Non-parametric approaches can be applied to a wide variety of textures (with irregular texture patterns) and provide better perceptual results (higher Mean Opinion Score (MOS) values). However, these schemes are often computationally more complex.

Recent work in the parametric approach is typically based on Auto Regressive (AR) [3] or Auto Regressive Moving Average (ARMA)-based modelling [4, 5, 6]. AR- and ARMA-based models in texture synthesis enable blocks to be selectively removed at the encoding stage, and reconstructed at the decoder with acceptable perceptual quality. AR- and ARMA-based approach are suitable for the textures with stationary data, like steady water, grass and sky, however they are not suitable for structured texture with non-stationary data (e.g., *brickwall*: Fig. 8) as blocks with non-stationary data are not amenable to AR modelling. Further, they are block-based approaches, and blocking artifacts can appear in the synthesised image.

Non-parametric approaches are pixel-based or patch-based. Efros and Leung [7] proposed pixel-based non-parametric sampling to synthesize texture. Wei and Levoy [8] further improve the above using a multi-resolution image pyramid based on a hierarchical statistical method. A limitation of the above pixel-based methods is an incorrect synthesis owing to incorrect matches in searching for close-by similar statistics. Patch-based methods overcome this limitation by considering features matching patch-boundaries with multiple pixel statistics. People generally use Markov Random Field (MRF) models for texture analysis [8, 9]. The popular choice for texture synthesis is a patch-based graph cut [10, 11]. Recent region based texture representation and synthesis algorithms [12] address the limi-

tations of the block based representation in handling homogeneous textures and blocking artifacts.

Zhang et al. [13] propose a DEMD-based image synthesis approach with a small representative texture sample. This patch is subjected to a DEMD-based decomposition for efficient representation of the texture information. To reconstruct a full-level image, the authors start from a highest level (say, *k*) of the DEM decomposition of the representative patch. They construct the *k*-th IMF corresponding to the full-resolution size by taking information from smaller patches inside the sample (the authors show an example with 64×64 representative sample textures, and smaller 8×8 patches), and placing them in the template for the full-resolution IMF (with overlap, to enforce smoothness). For the next lower IMF($k - 1$th) onwards, the authors perform a correlation search for each smaller block in the full-resolution template, with the closest-matching smaller block in the $k - 1$th level decomposition of the representative texture sample. The synthesised full-resolution image is the DEMD synthesis of the full-resolution IMFs. This approach shows good results for textured image synthesis with perceptually acceptable quality. However this approach leaves the following important questions unanswered: (a) The selection criterion for the small representative texture: its size and location, (b) Selection criterion for the size of the smaller patch, and the extent of the overlap (10-20% of the patch size in examples in the paper), (c) objective assessment of the quality of the synthesised image, (d) the level of IMF decomposition required for optimal synthesis, (e) inability to handle texture sequences with irregular patterns, as the authors themselves mention [13]).

Our proposed scheme is a hybrid approach, combining the advantages of parametric and non-parametric methods, which enables us to handle a wide variety of textured videos which cannot be accounted for by either method alone. DEMD can handle textures with stationary and non-stationary data. The scheme is in an H.264-/MPEG-based framework, for better compatibility with existing video coding standards. We take advantage of the temporal encoding efficiency of standard video coding frameworks (H.264/MPEG) to encode the DEMD residues frames. The synthesis is done using correlation search and patch-based multi-level IMF synthesis from the DEMD representation of key frames (*I* frames, in the H.264/MPEG framework) and the DEMD residual image of the frame being synthesised. Unlike [13], we provide both compression results, as well as an objective (PSNR) and perceptual (MOS-based) assessment of our encoded results. We give an energy encoding efficiency-based heuristic to determine the number of decomposition levels in the DEMD, which can either be sent from the encoder to the decoder, or can be independently estimated by the decoder in the same way as the encoder did it, for the decomposition and subsequent encoding. Unlike the work of Zhang et al. [13], since we perform the DEMD-based analysis of the full resolution frame, this does not suffer from the limitations of choosing a small representative patch size (which anyway cannot be done efficiently for an irregular texture pattern). Working with a full resolution instead of image blocks (as in [13]) typically gives us more compression with an acceptable perceptual reconstruction quality.

To the best of our knowledge, no related work address these issues. A very preliminary work in this direction was our earlier work [14]. The current submission differs considerably from the above preliminary work in the following ways. The earlier work encoded the IMFs' maxima and minima for multiple IMF levels, instead of the more efficient residue encoding in the current work. Moreover, the temporal information encoding in the H.264/MPEG framework - is a major contribution of the current submission. A heuristic in the earlier work [14] was to drop the residue (assuming that the residue was not sufficiently informative), leading to comparatively worse reconstruction quality, as well as compression (since the maxima and minima of many IMFs were also encoded).

The rest of the paper is organized as follows: Section. 2, provides an overview of DEMD-based video encoding and decoding framework. Section. 3 provides details of our test set-up and experimental results followed by a conclusion in Section. 4.

2. DEMD-based Video Coding

Huang et al. [15] propose Empirical Mode Decomposition (EMD) as a decomposition of a signal $f(t)$ into a 'low frequency/residue' term $r_1(t)$ and a 'high frequency/detail' part $i_1(t)$. Each $i_k(t)$ can be similarly decomposed, to give a K-level decomposition,

$$f(t) = \sum_{k=1}^{K} i_k(t) + r_K(t) \tag{1}$$

Here, functions $f_k(t)$ are the Intrinsic Mode Functions (IMFs) [15], [16], [17]. Advantages of such an iterative decomposition include the ability to handle any class of signals, non-linear, linear, non-stationary, and stationary. Further, the decomposition at each level is a simple numerical technique, and has an associated concept of a local scale (of an oscillation), and involves a perfect reconstruction. Liu et al. [18], [19], [20] extend the basic idea to 2-D, considering a dominant image direction (which is natural for an image segment containing texture) θ:

$$f(x, y) = \sum_{k=1}^{K} i_k^{\theta}(x, y) + r_K^{\theta}(x, y) \tag{2}$$

θ can be estimated using a Wold decomposition-based method [21] using the maximum of the Radon transform of the spectrum [13]. For instance, Fig. 2 shows dominant texture direction extraction for the *brickwall* sequence used in our experiments. *This is typically not expected to change appreciably for a textured video.* In the following sections, we discuss how we use DEMD for our encoder and decoder framework.

2.1. DEMD-based Encoder Framework

We consider a video as being composed of a set of shots, where each shot has an I frame, followed by a sequence of non-I (P, B) frames. (One can have either a fixed number of such non-I frames, or have this varied adaptively according to the motion in adjacent frames, and the associated propagation error.) Fig. 3 gives an overall block diagram of the proposed encoder. An I frame is sent to an H.264/MPEG encoder, which creates the

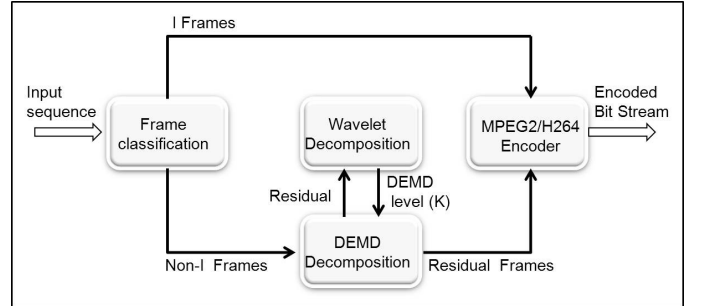**Fig. 2. The dominant texture direction, for the *brickwall* sequence.**

Fig. 3. The Encoder framework

encoded bit-stream. For a non-I frame to be encoded, the system creates the DEMD decomposition of the frame according to Eq. 2. We use a simple wavelet-based method to decide on the number of levels K for the DEMD process. We estimate K from the corresponding DEMD of the I frame.

The theory of EMD guarantees a decrease in the number of extrema across decomposition levels ($O(\log N)$ for an N-point function) [17]. Practically, one needs to stop when the information content in the residue is insignificant. This is often best done in the transform domain. We choose the computationally light Haar wavelets ($O(N)$) for an N-point signal). A study of the local properties (amplitude, phase, isotropy and orientation) of a 2-D IMF in the wavelet domain reveals that the HH sub-band energy best represent the IMF image, while the LL sub-band represents the overall entropic information. We compute the ratio of the energies in the HH and LL sub-bands, and stop the decomposition once this ratio becomes a small fraction of that for the original input frame. We selected this fraction empirically (0.01) based on experiments with a wide range of texture videos, and monitoring the synthesis quality (MOS as well as PSNR). Any further decomposition results in unnecessary computations with little tangible gain in reconstruction quality, or compression (e.g., Fig. 7).

The encoder sends the penultimate-level residues $\{r_{K-1}^{\theta}(x, y)\}$ *- the logic behind sending the $(K-1)$th level residue is to enable the decoder to have some information content to enable it to perform a search for the closest patch in the DEMD decomposition image at a particular level. (Sec. 2.2 has the details of the process.) The motivation behind this is the same as that for splitting frames of a video into I and non-I frames: the latter can be predicted (P: unidirectionally, or B: bidirectionally) from an I frame, due to temporal redundancy.* Hence, it is reasonable to assume spatial correlation of a kth level decomposition between an I and a non-I frame. For an I frame,

the input to the H.264/MPEG encoder is the frame itself. *For a non-I frame, the input to the H.264/MPEG encoder is the $(K-1)$ level residue, which is expected to contain far less information as compared to the original frame, to enable it to be more efficiently compressed using a standard H.264/MPEG encoder.*

2.2. DEMD-based Decoder Framework

The decoder receives the H.264/MPEG stream, with each shot segment composed of an *I* frame, and a set of non-I $(K-1)$th level residue frames $\{r_{K-1}^\theta(x,y)\}$. Fig. 4 shows a block level view of the proposed decoder framework.
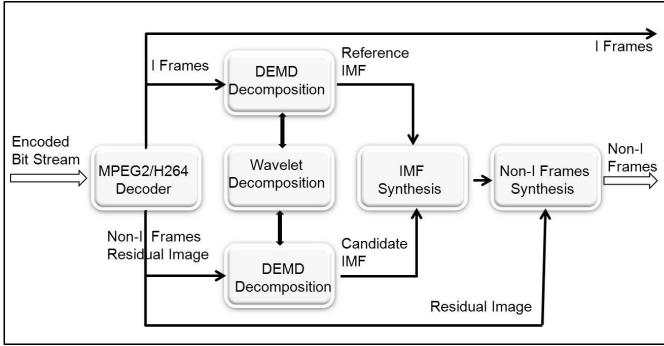

Fig. 4. The Decoder framework

The decoder first decomposes the decoded *I* frame into K DEMD components $i_k^{\theta_k,I}(x,y)$, according to the same energy-based stopping criterion as in the encoder (Sec. 2.1). The decoder also estimates θ exactly as the encoder does it. For every non-*I* frame, the decoder starts from the H.264/MPEG-decoded $(K-1)$th level residue frame. It performs an additional decomposition of $r_{K-1}^{\theta,non-I}(x,y)$ into $i_K^{\theta,non-I}(x,y)$ and $r_K^{\theta,non-I}(x,y)$. As mentioned in the previous section (Sec. 2.1), we assume that the decomposition proceeds till there is very little information content in the Kth level residue. The $(K-1)$th level residue is the last decomposition level with some useful information in the residue image $r_{K-1}^{\theta,non-I}(x,y)$. The decoder does a correlation-based search for each block in the target IMF image $i_{K-1}^{\theta,non-I}(x,y)$, with a block in the corresponding *I* frame IMF image $i_{K-1}^{\theta,I}(x,y)$, with an overlap.

Each pixel in the IMF image has a 3-tuple feature vector: the magnitude ('instantaneous bandwidth'), and the two angular directions (the 'instantaneous frequencies') [22, 13]. To get the best matching patch, we do a local search in two 8×8 blocks on each side of the candidate block in the upper level newly-synthesised IMF image (with overlap), and select the one with least Euclidean distance from the corresponding block in the *I* frame IMF. We repeat the procedure for the other decomposition levels as well, using the same correspondence information for the blocks from the correspondence information at level $(K-1)$. Fig. 5 demonstrates the overall block level view for IMF decomposition and synthesis flow described above.

2.2.1. Patch Selection & Synthesis: Issues, Workable Solutions

It may be argued that an EMD generates IMFs with overall decreasing frequency content, so one may pose a question of
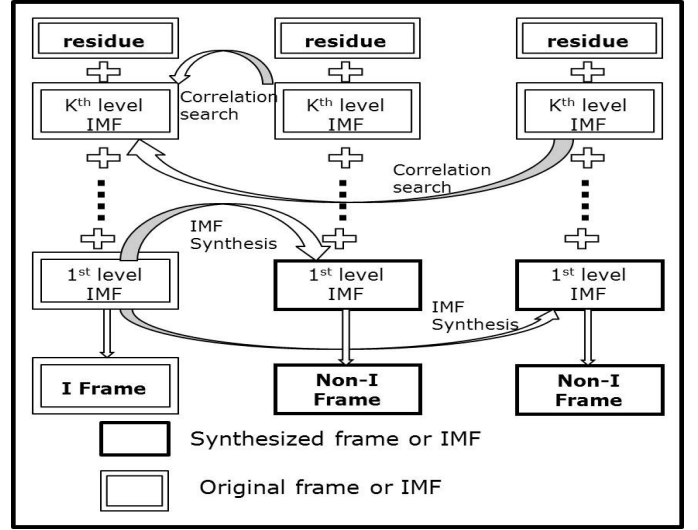

Fig. 5. The IMF Decomposition and synthesis flow

having an adaptive block size for the block texture map. However, EMD involves the concept of a local frequency and a local scale [15], [16], [17], and it is not easy to have an adaptive block size.

As a compromise, for a texture synthesis application [13], Zhang et al. use a fixed block size, but perform correlation search with the immediate previously synthesised IMFs (assuming that the frequency content will not change drastically between two adjacent IMF levels - an approximation in terms of the theory, but a workable heuristic). An even more gross approximation is a 'flat search': to assume that the correspondence of blocks determined at the Kth IMF level, remains unchanged across other IMF levels also. The advantage of this gross approximation is less computational cost, since we do not have to work out a correspondence across multiple IMF decomposition levels. The trade-off is the reconstruction quality. The limitation is usually visible in artifacts across the patch boundary of the synthesised IMFs.

As a workable solution, we assume an empirically chosen compromise block size, and perform a correlation search across all prior synthesised IMFs for each block, at the cost of increased computational complexity, but to ensure that most localised frequency transitions can be captured across multiple decomposition levels (for the cases where a correlation search with the immediately previous synthesised IMF does not capture the best possible block correlation). Fig. 6 shows a comparison of a 'flat search' with correlation search, for two and three levels of IMF decomposition, for instance. for six representative videos.

The approach in this paper works across temporal redundancy as well (in a video with frames). Even at the level of a frame, our approach differs from that of Zhang et al. [13] as we encode full-resolution reference data, unlike a small set of sample texture (whose size is empirically selected as 64×64 and content is expected to represent all the structural texture unit in the image). This enables us to extend our proposed framework for a wide variety of textures, such as irregular texture patterns.
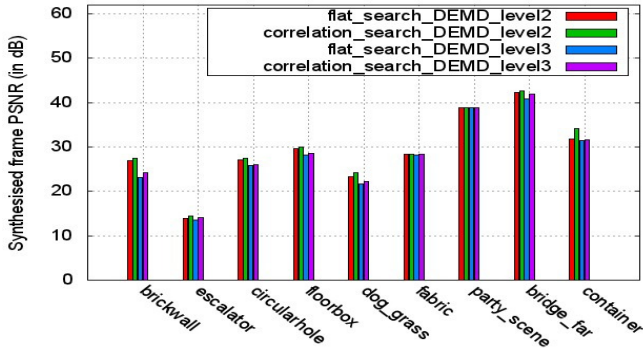
Fig. 6. Illustration of patch based synthesis using a 'flat search' versus a correlation search (Sec. 2.2.1 has the details and explanation). Correlation search based patch synthesis provides better simulation results (PSNR data) as compared to 'flat search'-based patch synthesis.

Fig. 7. Compression gain over MPEG2/H.264 for the proposed multilevel IMF synthesis

The synthesis is also faster in our proposed scheme because it uses full resolution reference data: finding the best possible patch requires a very localised search, unlike the exhaustive search in case of sample based texture synthesis scheme [13].

For the synthesised texture quality assessment, we do objective quality assessment using PSNR as well as subjective assessment using MOS, while Zhang et al. [13] use only a subjective assessment criterion and rate the reconstruction quality with the visual effect achieved.

3. Results and Discussion

In this section, we present experimental results for representative structured textures (Sec. 1) with periodic (*escalator*, *fabric*) and pseudo-periodic (*brickwall*, *party_scene*), and non-structured/statistical textured sequences (*floorbox*, *circularhole*, *dog_grass*, *bridge_far*, *container*). All the test sequences are 128×128, 8-bit, gray scale videos. PSNR and MOS have been used as the quality metrics. Mean Opinion Score(MOS) computation was done by collecting responses of various students and staff working in the lab and averaging them. All the experiments are carried out in MATLAB, running on a Windows XP PC with a P4 CPU and 1GB RAM.

Table 1 shows sample compression results with just a single level of IMF decomposition. The H.264 compression is larger than the MPEG2 one (better spatial and temporal redundancy encoding), and our scheme performs significantly better than either, in the respective frameworks. An exception to the above trend is for the *brickwall* and *container* sequences, where the significant periodicity with high structural transitions between frames in *brickwall* and dynamic statistical transition between frames in *container* sequence, limits the compression gain with DEMD as opposed to encoding the raw video data. This is a limitation of the proposed scheme, as summarised towards the end of this section.

The compression gain increases considerably with the number of levels of IMF decomposition, as dictated by the energy-based decomposition stopping criterion (Sec. 2.1). Fig. 7 shows some representative results. The importance of the energy-
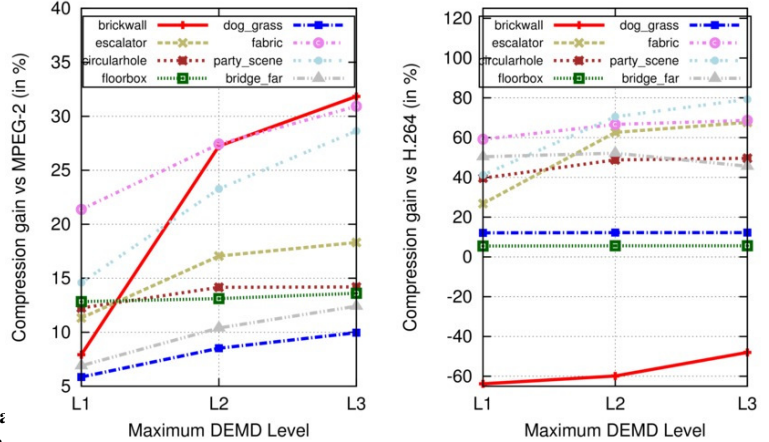
based stopping criterion can be noted here, as some test sequences as *floorbox* have an optimal compression with acceptable quality, at the first IMF decomposition level itself. Additional levels of IMF decomposition for such sequences only add computational complexity without any significant gain.

Figs. 8, and 9 demonstrate synthesis results of the four textured video sequences comprising synthesised IMF image and non-*I* frame sequences. All images with framed boundaries are synthesised frames. (The decoder performs one level of decomposition to get the *K*th level residue and the *K*th IMF: details in Sec. 2.2). The Mean Opinion Score(MOS) data on a scale of 5 indicates that the synthesis quality is perceptually acceptable with a comparable PSNR range to H.264, as shown in Table 1.

It is interesting to note that in the experiments with the dog_grass *sequence, we have not explicitly performed a foreground-background separation, and encoded the background (the grass, and the rocks in the meadow) alone. In spite of this, the proposed framework works better as compared to a standard MPEG2/ H.264 encoding, as can be seen from the data in Table 1, and the graphs in Fig. 7 even for one IMF level alone, and more so for further levels of IMF decomposition.*

The approach in this paper combines the effective decomposition property of DEMD to exploit temporal redundancy, within the framework of the H.264/MPEG standard for flexibility with regard to a standard, and a possibly better feature representation scheme for textured videos. A limitation of the proposed scheme is in cases such as the *brickwall* and *container* videos, where the H.264 temporal redundancy representation for the DEMD residue does not give an appreciably better compression gain as compared to working with the raw frame data. Such situations typically arise when the video data has considerable aperiodicity with high structural transitions or dynamic statistical transition between frames.

4. Conclusion

In this paper, we propose a DEMD-based textured video coding scheme for efficient representation of textured data in a

Table 1. Even a single level of IMF decomposition provides considerable compression gain over MPEG2 and H.264, at acceptable quality levels (PSNR as well as MOS) - for representative structured (Sec. 1) as well as non-structured/statistical textures. Sec. 3 has the details and the explanations. Columns 2 & 3 give the encoded bit sizes using MPEG-2 alone, and the DEMD-based proposed scheme in an MPEG-2 framework ('proposed In MPEG2'). Columns 5 & 6 show the corresponding figures for H.264.

Sequence	Encoded BitSize standard MPEG2	Encoded BitSize proposed In MPEG2	%Gain Over standard MPEG2	Encoded BitSize standard H.264	Encoded BitSize proposed In H.264	%Gain Over standard H.264	Quality (PSNR,dB) standard H.264	Quality (PSNR,dB) proposed In H.264	Quality (MOS) standard H.264	Quality (MOS) proposed In H.264
circularhole	37220	32659	12.25	6940	4190	39.63	36.00	35.03	4.5	4.4
escalator	43806	38849	11.32	12188	8920	26.81	34.00	30.16	4.5	4.4
floorbox	36268	31613	12.83	3632	3442	5.50	37.00	37.80	4.4	4.2
dog_grass	24408	22977	5.86	3176	2792	12.09	35.84	34.64	4.5	4.2
fabric	53514	42074	21.38	20726	8460	59.18	32.59	28.43	4.5	4.0
party_scene	51707	44165	14.59	12544	7376	41.20	35.06	39.31	4.5	4.6
bridge_far	28778	26792	6.90	2932	1406	50.35	36.49	44.34	4.6	4.7
container	35682	33120	7.18	1322	2902	-54.44	37.32	33.47	4.5	4.2
brickwall	49017	45133	7.92	1516	4192	-63.08	34.43	39.01	4.5	4.5

H.264/MPEG framework. We encode the DEMD residue image of all non-*I* frames (*P* and *B* frames in H.264/MPEG standard) in a H.264/MPEG framework, and not the original frames themselves. All non-*I* frames are synthesised using correlation search and patch-based multi-level IMF synthesis from the DEMD representation of key frames (*I* frames, in the H.264/MPEG framework) and the DEMD residual image of the frame being synthesised. Experimental results show that the proposed scheme provides better compression results as compared to MPEG/H.264, with acceptable reconstruction quality. A possible application of the proposed framework could have been for efficient representation of a BRDF (Bi-Directional Reflectance Function) of an object, where different images of an object in a single pose are taken in different illumination conditions. A video representation of the same may offer better compression in representation as compared to storing a large set of such images. If such a dataset has significant structural textural information, an approach similar to the proposed approach may be viable. Another possible application scenario could be hyper-spectral satellite imagery, where images in any band typically contain textured areas. Representing these textures in different bands as a video using the proposed DEMD-based framework of this paper could be a possible efficient representation.

References

[1] ITU-T Rec. H.262 and ISO/IEC 13818-2 MPEG-2, Generic Coding of Moving Pictures and Associated Audio Information - Part 2 Video.

[2] ITU-T Rec. H.264 and ISO/IEC 14496-10 (MPEG-4/AVC), Advanced video coding for generic audio visual services, Standard version 7, ITU-T and ISO/IEC JTC 1.

[3] A. Khandelia, S. Gorecha, B. Lall, S. Chaudhury, M. Mathur, Parametric Video Compression Scheme using AR based Texture Synthesis, Proc. IAPR- and ACM-sponsored Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP) (2008) 219 – 225.

[4] A. Stojanovic, M. Wien, J. R. Ohm, Dynamic texture synthesis for H.264/AVC inter coding, Proc. IEEE International Conference on Image Processing (ICIP) (2008) 1608 – 1612.

[5] A. Stojanovic, M. Wien, T. K. Tan, Synthesis-in-the-loop for video texture coding, Proc. IEEE International Conference on Image Processing (ICIP) (2009) 2293 – 2296.

[6] H. Chen, R. Hu, D. Mao, R. Thong, Z. Wang, Video coding using dynamic texture synthesis, IEEE International Conference on Multimedia and Expo (2010) 203 – 208.

[7] A. Efros, T. Leung, Texture synthesis by non-parametric sampling, International conference on computer vision (1999) 1033 – 1038.

[8] L. Y. Wei, M. Levoy, Fast Texture Synthesis using tree-structured vector quantization, in Proc. SIGGRAPH, New Orleans, Louisiana, USA (2000) 479 – 488.

[9] P. Ndjiki-Nya, M. Koppel, D. Doshkov, T. Wiegand, Automatic structure-aware inpainting for complex image content, International Symposium on Visual Computing (2008) 1144 – 1156.

[10] V. Kwatra, A. Schodl, I. Essa, G. Turk, A. Bobick, Graphcut textures : image and video synthesis using Graph Cuts, in Proc. SIGGRAPH, San Diego, CA, USA (2003) 277 – 286.

[11] S. Ierodiaconou, J. Byrne, D. R. Bull, D. Redmill, P. Hill, Unsupervised image compression using graphcut texture synthesis, Proc. IEEE International Conference on Image Processing (ICIP) (2009) 2289 – 2292.

[12] A. Dumitras, B. G. Haskell, An Encoder-Decoder Texture Replacement Method with Application to Content-Based Movie Coding, IEEE Transactions on Circuits and Sytems for Video Technology 14 (2004) 825 – 840.

[13] Y. Zhang, Z. Sun, W. Li, Texture Synthesis based on Directional Empirical Mode Decomposition, Computer and Graphics (2008) 175 – 186.

[14] M. K. Jha, B. Lall, S. Dutta Roy, Video Compression Scheme Using DEMD Based Texture Synthesis, in: Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG), 2011, pp. 90 – 93.

[15] N. E. Huang, Z. Shen, S. R. Long, The Empirical Mode Decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proceedings of Royal Society of London (Series A) 1 (1998) 903 – 995.

[16] A. Linderhed, Empirical Mode Decomposition as Data-Driven Wavelet-Like Expansions, Ph.D. thesis, Dissertation No. 909, Linkoping University (2004).

[17] P. Flandrin, P. Goncalves, Adaptive Image Compression with Wavelet Packets and Empirical Mode Decomposition, International Journal of Wavelets, Multiresolution and Information Processing (2004) 1 – 20.

[18] Z. Liu, H. Wang, S. Peng, Texture classification through Directional Empirical Mode Decomposition, Proc. International Conference on Pattern Recognition (ICPR) 4 (2004) 803 – 806.

[19] Z. Liu, H. Wang, S. Peng, Texture Segmentation using Directional Empirical Mode Decomposition, Proc. IEEE International Conference on Image Processing (ICIP) 1 (2004) 279 – 282.

[20] Z. Liu, S. Peng, Directional Empirical Mode Decomposition and its application to Texture Segmentation, Science in China Series E: Information Sciences 2 (2005) 113 – 123.

[21] J. Francos, A. Meiri, B. Porat, A unified texture model based on a 2-d wold-like decomposition, IEEE Transactions on Signal Processing 41 (1993) 2665 – 2678.

[22] T. Bulow, G. Sommer, Hypercomplex signals- a novel extension of the analytic signal to the multidimensional case, IEEE Transactions on Signal Processing 49 (2001) 2844 – 2852.
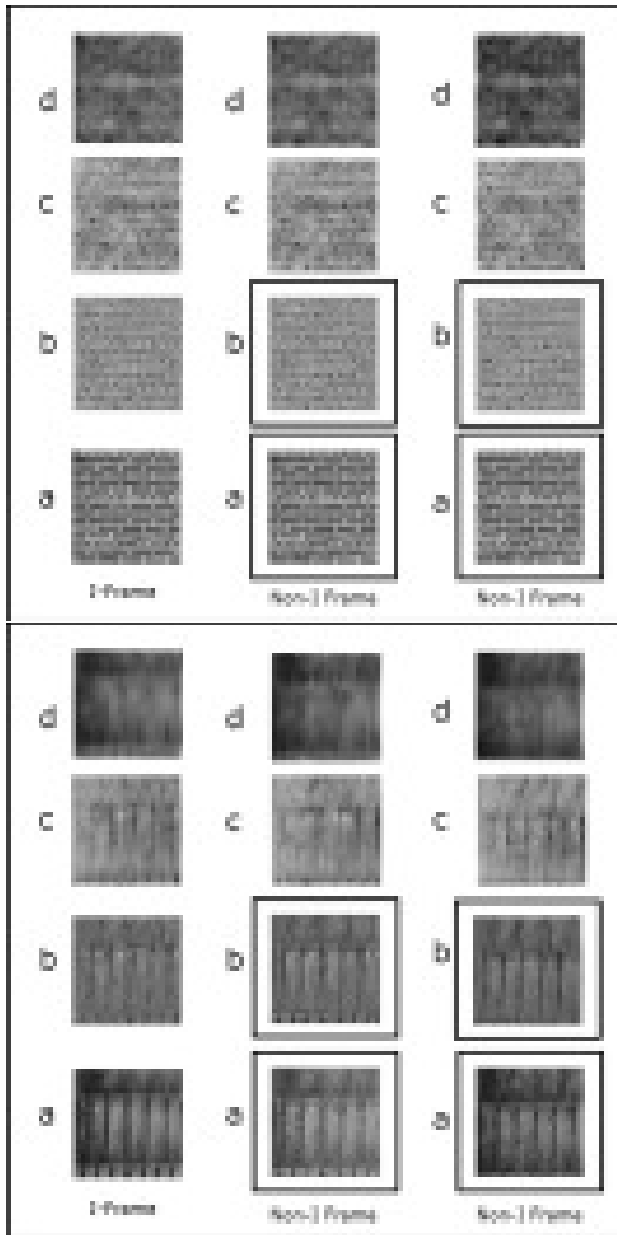
Fig. 8. Synthesis results for sequences (in raster-scan order), *brickwall* and *escalator*: a. Input *I* frame and synthesised non-*I* frames, b. 1st level IMF image for *I*-frame and 1st level synthesised IMF image for non-*I* frames, c. 2nd level IMF image, and d. DEMD residue image. The images with framed boundaries are synthesised frames: the decoder decomposes the encoded $(K-1)$th level residue into the Kth level residue and IMF, and performs a correlation search to get the synthesised frames: Details in Sec. 2.2, Sec. 2.2.1.
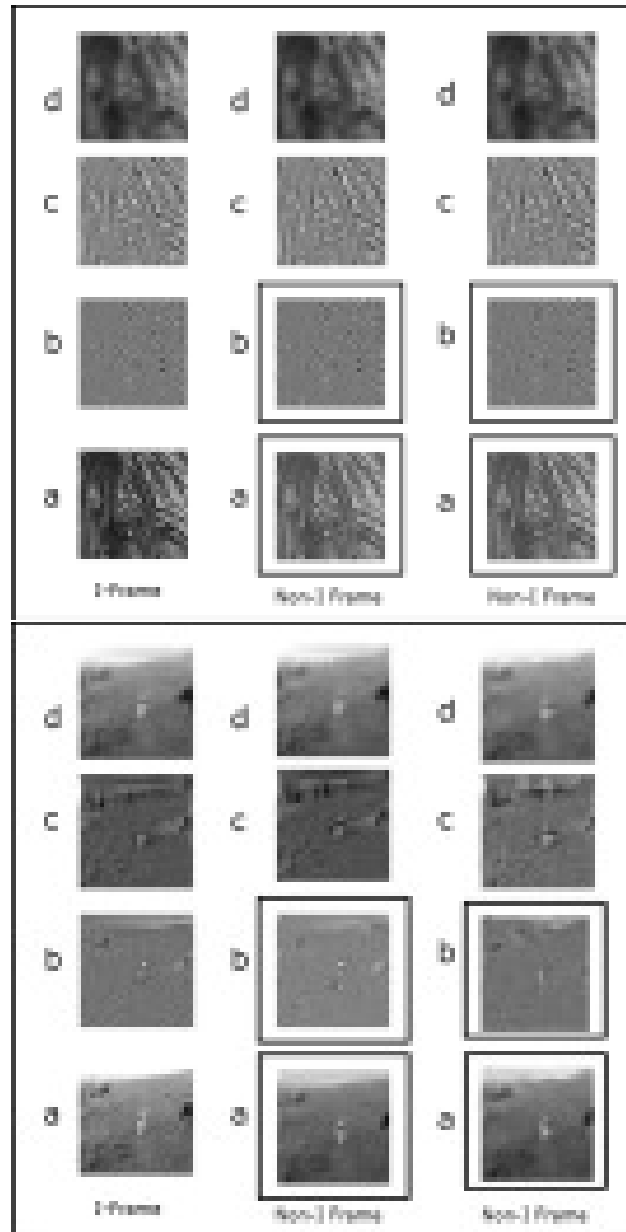

Fig. 9. Synthesis results for sequences *party_scene* and *dog_grass* : a. Input *I* frame and synthesised non-*I* frames, b. 1st level IMF image for *I*-frame and 1st level synthesised IMF image for non-*I* frames, c. 2nd level IMF image, and d. DEMD residue image. As mentioned in the text (Sec. 3), it is interesting to note that the sequence *dog_grass* is not a pure 'textured' video and we have not extracted the foreground (the dog) and coded only the background texture. In spite of this, Table 1 and Fig. 7 show that the compression in the proposed framework is better than a standard MPEG2/H.264 encoding. The images with framed boundaries are synthesised frames: the decoder decomposes the encoded $(K-1)$th level residue into the Kth level residue and IMF, and performs a correlation search to get the synthesised frames: Details in Sec. 2.2, Sec. 2.2.1.