Video compression by Neural Network

Vigliano Daniele; Raffaele Parisi; Aurelio Uncini

Università degli Studi di Roma “La Sapienza” Via xxxxx

Introduction

“A picture is worth a thousand words”. This popular saying well synthesizes the different relative importance between visual and textual or linguistic information in everyday’s life. As a matter of fact, visual information has reached an essential and undisputed role in modern Information and Communication Technology. In particular, the widespread diffusion of telecommunications and networking today offers new opportunities to the transmission and processing of multimedia data. Nevertheless, the transmission of highly informative video contents imposes strict requirements in terms of band occupancy. A trade-off between quality and compression is thus asked for.

Specifically, compression of video data aims at minimizing the number of bits required to represent each frame image in a video stream. Video compression has a large number of applications in several fields, from telecommunications (teleconferencing, e-learning), to remote sensing, to medicine. Depending on the application, some distortion can be accepted in exchange for a higher compression ratio. This is the case of so-called lossy compression schemes. In other cases (e.g. biomedical applications), distortion is not allowed (lossless coding schemes).

Video compression techniques have been classified [45] into four main classes, according to the distinction among waveform, object-based, model-based and fractal coding techniques.

Waveform compression techniques refers to temporal axis as a third dimension, belong to this category all the application working in time domain as DCT and Wavelet but also Motion compensation techniques [58][CCITT2]. *Object based techniques* considers video sequence as a collection of different objects [62] that can be differently processed, they are extracted by a segmentation step [44]. *Model based* approach perform the analysis of the video and the synthesis of a structural 3D or 2D model [66]. *Fractal Based techniques* extends to video applications the success reached in image coding; an image can be expressed as the attractor of a contractive function system and then retrieved by iterating the set of function [73]. Several standards have been also developed.

In last years there has been a tremendous growth of interest in the use of neural networks for video coding. This interest is justified by the well-known capabilities of neural networks of performing complex input-output nonlinear mappings, in a learning from examples fashion. Neural Network improves the performance of all the four compression techniques.

This chapter gives an overview of the major neural technique already used and detail one of that. It is organized as follows. In section “Review of recent standards” a short description of most recent standards in video compression is provided. Section “Neural video compression: existing approaches” presents an overview of most popular neural approaches to video coding, while Section “Neural video compression the image coding approach” describes two specific and particularly effective solutions.

Review of recent standards

Image and video have been the object of intensive research in the last twenty years. The diffusion of a large number of compression algorithm leads to the definition of several standards; two international organization (ISO/IEC and ITU-T) have been heavily involved in standardization of images, audio and visual data. To have a complete overview of recent standard and recent trends in visual information compression see [45][51][52]; a detailed description of the standard here addressed is out of the scope of this section, more details can be found in the references.

The standards proposed for general purpose still images compression are the JPEG [46][47] based on a block DCT transform followed by an Huffman or Arithmetic coding, and the more recent JPEG2000 [48][49][50] based on discrete wavelet transform and EBCOT coding.

On the video compression side, hybrid schemes that reduce the spatial redundancy by DCT and temporal correlation by motion compensated prediction coding are used in ITU H.261 [53]. It was designed and optimized for videoconference transmission over an ISDN channel (a bit rate down to 64 kbit/sec).

H.263 [56] and H.263+ [54] have the same core architecture of H.261 but some improvements are introduced principally in precision of motion compensation and in prediction; they allow the transmission of audio video information with a very low bit rate (9.6 Kb/sec).

Last advances in video coding aim at collecting all the suitable feature previously used in video compression to develop new standards (still in developing) that outperform all the just introduced. One of this new algorithm is the H.26L [77][55].

The first studies of the Moving Picture Expert Group (MPEG) starts in 1988, they aim at developing new standards for the Audio Video Coding. The main difference with respect to the other standards is that MPEGs are “open standard”, so they are not dedicated to a particular application.

MPEG-1 was developed to operate at bit rates of up to about 1.5Mbit/sec for the consumer video coding and video content store on media like CD ROM, DAT; it provides important features including frame based random access of video, fast forward/fast reverse (FF/FR) searches through compressed bit streams, reverse playback of video and editability of the compressed bit stream. MPEG-1 perform the compression using several algorithms such as the subsampling of video information to match the HVS (human video system), variable length coding, motion compensation and DCT to reduce the temporal and spatial redundancy [57][58][59].

MPEG-2 is similar to MPEG-1 but it include some extensions to cover a wider range of applications (e.g. HDTV and multi channels audio coding). It was designed to operate at a bit rate between 1.5 and 35 Mb/sec. One of the main enhancement of MPEG-2 over MPEG-1 is the introduction of syntax for efficient coding of interlaced video. The Advanced Audio Coding (AAC) is one of the formats defined in the non back-compatible version of MPEG-2; it was developed to perform the multichannel audio coding. The MPEG-2 AAC is based on the MPEG-2 layer III, some blocks are improved (frequency resolution, joint stereo coding, the Hoffman coding) and some others like spectral and time prediction was introduced. The resulting standard is able to perform the coding of five audio channel [60][61].

From the evolution of the object oriented computer science comes the use of objects into video compression; this leads the developing of MPEG-4: the video signal can be considered as composed by different objects, with theirs own shape, motion and texture representation. Objects are coded independently in order to allow direct access and manipulation. The power of this coding approach is that different objects can be coded by different tools with different compression rate; in a video sequence some parts of the scene could require less distortion but some other not. The original video is than divided in streams: audio and video stream are separated, each object have its own stream, as information about object placement, scaling and motion (Binary Format of Scene).

In MPEG-4 Synthetic an Natural sounds are coded in a different ways, the Synthetic Natural Hybrid Coding (SNHC) perform the composition of natural compressed audio and of synthetic sounds (artificial sound are created in real time by the decoder); MPEG-4 proposes also the division between speech and “non speech” sound because the first one can be compressed by ad hoc techniques [62][64][63][65].

In last years the value of information starts to became not only the information itself but how easy one can access, manage, find, and filter such information. MPEG-7 formally named “Multimedia Content Description Tool” provide a rich set of tool performing the description of audio-visual content in multimedia environment. The application areas which benefit from audio-video content description are in different fields: from the web search of multimedia content to the broadcasting media selection, from the cultural services (like art gallery) to home entertainment, from the journalist application to the more general databases (of multimedia data) applications [67][68][69][70]. The descriptions provided by MPEG-7 are independent of the compression method. Descriptions have to be meaningful just in the context of the considered application, for this reason different types of features perform different abstraction levels. MPEG-7 standard consist of several parts, in this section Multimedia Description Schemes, the Visual description tool and the Audio description tool are detailed

Multimedia Description Schemes (DSs) are metadata structures to describe audio-visual content, it is defined by the Description Definition Language (DDL) based on XML. Resulting descriptions can be expressed in text form (TeM) or in a binary compressed form (BiT); if the first one allow human reading and editing, the second one improve the efficiency in storing and transmission. In this framework are developed tools providing DSs with information about the content and the creation of the multimedia document and DSs to improve the browsing and the access to the audio-visual content. Visual description tool performs the description of visual category like colour, textures, motion, localization, shape and face recognition. Audio Description tool contains low level tool (e.g. Spectral, temporal audio feauters descriptions) and high-level specialized tool like musical instrument timbre, melody description, spoken tools and the one for the recognition and indexing of general sound. MPEG-7 standard provides also an application to represents the multimedia content description named “Terminal”; it is important to underline that the Terminal performs both the downstream and the upstream transmission involving less or more specific queries from the end user.

The MPEG standards just introduced are interested in processing of the multimedia content in a physical context and in a semantic one (MPEG-7); they does not addresses other issues like multimedia consumption, diffusion, copyright, access or management rights. Based on the above observation MPEG-21 aims at resolving that lack by providing new solutions to access, consumption, delivery, management and protection process of the different content types.

MPEG-21 is essentially based on two concepts: Digital Item and Users. The Digital Item (DI) represents the fundamental unit of distribution and transaction (e.g. video collection, musical album); it is modelled by Digital Item Declaration (DID): a set of abstract terms and concepts. The Users are every entity (e.g. humans, communities, society) that interacts with MPEG-21 environment or uses Digital Items. The management of Digital Items are allowed by the User right to perform the action [71][72].

Neural video compression: existing approaches

This section introduces some interesting neural applications. Neural networks in video compression can be used in two main ways: as a stand alone method or as a part of algorithm.

In this last context NNs introduces improvements in coding schemes of intra frame coding, in clustering capability, motion estimation and objects segmentation; the power of NNs as trained system is applied also in removing artifacts and post processing.

Important issue in video compression is the computational complexity that produces more complex physical implementation of the algorithm. As a matter of fact Artificial Neural Networks are usually computationally less expensive with respect to other algorithms, this is one of the reasons of the success of the neural network in video coding. .

In the following sections will be introduced application on Vector Quantization, Singularity Map creation and human vision approach, Motion Compensation and Fuzzy segmentation; these section aim at introducing some of the more representative existing approach of neural video compression.

Vector quantization

Vector Quantization is a very popular and efficient method for frame image (or still image) compression, it represents the natural extension of scalar quantization to n-dimensional space [17][18][19].

Figure 1 represents a conceptual scheme of a Vector Quantization coder, given a codebook, input vectors are quantized to the closest codeword so the output of the coder is the index of the codeword. Codebooks are generated using clustering algorithms from a set of training images. In [20] this optimization problem is approached by a Kohonen neural network with the same number of neurons as the number of pixel block; in such context the number of cluster (output neurons) are set to the desired numbers of codeword.

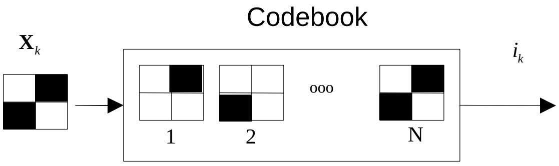


Fig. 1. Conceptual scheme of Vector quantization

The learning of the network is based on the evaluation of the minimum distance between outputs and inputs: the winner is the neurons with the lower value of that distance. The main advantages in using SOFM with respect to other clustering algorithm (k-means, LBG) include less sensitivity to initialization, better rate distortion performance and faster convergence; moreover SOFM during the learning grants to update not only the winning class but also the neighboring one, this because diminishing the chance of winning the competition produce the under-utilization of neurons.

For more details about the motivation that inspire the use of Self-Organizing feature map into the codebook designing see[21][22][20]. Suitable properties of SOFM can be used in performing more efficient codebook design, example are APVQ (Adaptive Prediction VQ), FSVQ (Finite State VQ) and HVQ (Hierarchical VQ).

APVQ uses ordered codebooks in which correlated input are quantized in adjacent codewords; an improvement in coding gain is obtained by encoding such codebook index with a DPCM (or some other neural predictor) [23].

FSVQ [24][Foster] introduces some form of memory in static VQ. It defines states by using the previously encoded vectors, in each state the encoder selects a subset of codeword of the global codebook; the Side Match FSVQ [29] in which the current state of the coder is given by the closer side of the upper and left neighboring vector (the block of the frame image).

In order to perform reduced computational effort, hierarchical structure can be used. In literature are widely diffused techniques that cascade the VQ encoders in several ways: two layers structure or hierarchical structures [27] based on topological information [26] .

In Vector Quantization framework other Neural applications are the two step algorithms; in [28] it was proposed an algorithm in which a neural PCA produces inputs for SOFM performing the VQ.

Singularity map and human vision

In several fields of audio-video processing the most interesting successes are reached by emulating the still better processing system: the human brain.

The approach on video compression introduced in this section is used in cases of very low compression ratio (about 1000:1) and it is inspired by the human vision system. For its own physiological structure it does not pay attention to each single pixel of an image or a video stream rather to the intensity changes. Focusing on particulars that are really important for the human perception provides a better quality in high compression situation; such particulars are edges and intensity changes.

At entering the light in human eye it focus on the retina in which there are two kind of receptors, rods and cones; retina uses the rows for monochromatic light and cones for the colour vision (RGB). Each receptor fires when it receives light, in firing it takes the resources from the near receptor in such a way to allow them a smaller excitation. So the dark areas became darker and the light area became lighter. This phenomenon inspired some artificial neural structure; it is well known as “lateral inhibition”. For these reasons retina is able to better detect edges than smooth surfaces. The transmission through the optical nerve suffers of propagation dispersion, this produce the smoothing of the edge by broadening the borders.

The Human Vision System (HVS) is the main difference between the approach here introduced and other approaches, for more detail see [31][30][32].

The algorithm is composed by two main parallel steps:

- very low bit rate compression performed with a method that does not produces block effect.
- singularity map computed from the original video, before the compression.

the final step correspond to the application of singularity map on compressed frames. A block scheme of the proposed technique is represented in figure 2. Results reached by the SM application are better than the one obtained with only the video compression techniques (upper path in figure 2).

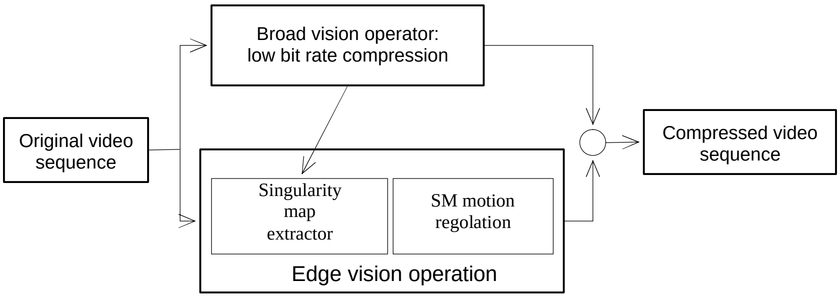


Fig. 2. Block scheme of the Human Vision System based compressor.

The algorithm performs two type of Singularity Maps: the Hard one for daily video sequences and the Soft one for nightly video sequences; moreover it takes in particular consideration the presence of noise into the original video sequence because it can produce a more difficult estimation of singularity map.

Singularity Map is obtained labelling, with topological index and greyscale correspondence, the singular point of the border of the frame image. By this way the whole edge can be transmitted as a sequence instead of as an image.

Singularity Map is the collection of the multiresolution edges of a frame image, the extraction processing requires special cares because ordinary edge extractors, like Sobel, broadens the edges map.

For Hard Singularity Map [31] proposes the use of iterative min-max, for the Soft SM it proposes the CNN (Cellular Neural Networks) that can extract sharpen edge in almost real time.

Once computed the SM the very low bit rate video compression is performed using EPWIC (Embedded predictive Wavelet Image Coder [33]), EZW [34] or other performing wavelet compression techniques.

Motion compensation

Motion compensation (MC) is one of the most performing techniques to reduce temporal correlation between adjacent frames. It is based on the issue that adjacent frames can be very similar so highly correlated in a large number of general purpose video applications. In order to reduce this correlation one block in a frame can be coded as a translated version of one block in a precedent frame, but have to be transmitted the motion vector too. In this framework only translational motion is considered.

In motion estimation framework, frames are segmented in macroblock of 16 x 16 pixes composed by 4 block of 8x8 pixels (a reduced block representation error correspond to finer block but it produce computational overhead). Figure 3 shows how in coding the block of frame *k* is computed the “best match block” of previous frame and than the representation error is coded together with the information of the “motion vector”

Several methods have been investigated in order to reduce the estimation error and in order to fasten the best match research; the so called predictive methods perform the matching research only towards previous frame, the bidirectional one consider also the future frames to perform a bidirectional estimation

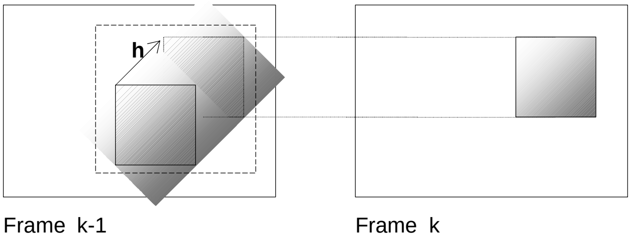


Fig. 3. Motion Compensation

In [35] is proposed an Hopfield neural algorithm to perform hierarchical motion estimation. It is used a classical best match method in order to reduce the number of possible macroblocks then, once obtained a subset of *D* candidates an Hopfield Network is used to obtain the best vector of affinities **v**. The optimum affinity vector **v** is the one that optimize the following equation

$$\frac{1}{2}\|\mathbf{f}-\mathbf{G}\mathbf{v}\|^2 = \frac{1}{2}\sum_{p=1}^{L}\left(f_p - \sum_{i=1}^{D} g_{p,i}v_i\right) \tag{1.1}$$

In equation (1.1) **f** is the vector of the current block to be estimated, **G** is a matrix which column are the *D* candidate block (produced by the first step) **v** is the affinity vector: the one that select the best match block.

Architecture of the neural network which perform the vector optimisation is represented in figure 4.

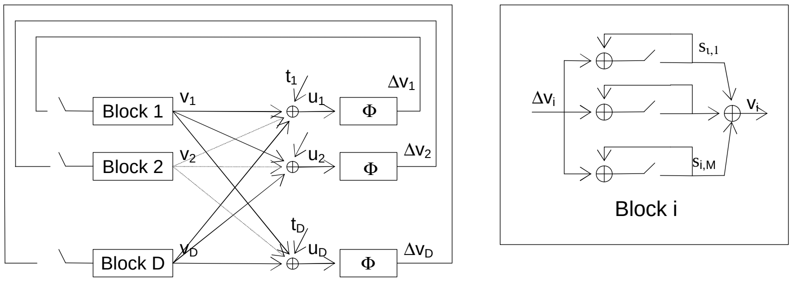


Fig. 4. Hopfield neural network to perform the motion estimation.

Other approaches on motion estimation by neural networks are performed by cellular neural network (CNN)[36][38][39][37][76].

These works aim at parallelize the computational flow required by both motion estimation and compensation; the application of CNNs perform faster and scalable computations.

Figure 5 shows the cell of the network presented in [36]; it graphically represent the set of difference equation given by:

$$C\dot{x}_{ij}(t) = \frac{x_{ij}(t)}{R} + \sum_{k,l} A_{i,j;k,l} y_{kl}(t) + \sum_{k,l} B_{i,j;k,l} u_{kl}(t) + I \tag{1.2}$$

in which the letter of the equation refers to the block of figure 5.

In [36] motion estimation is based on maximization of the a-posteriori probability of the scene random field given the random motion field realization; CNN are used because it has the same structure of the energy function of the network.

Cellular neural networks were designed to perform an optimization process based on its intrinsic property to evolve towards a global minimum state. Detail about algorithm, stability and network design can be found in [36][37][40].

The distributed computation capability, based on the parallel structure of CNNs, are used in other contexts.

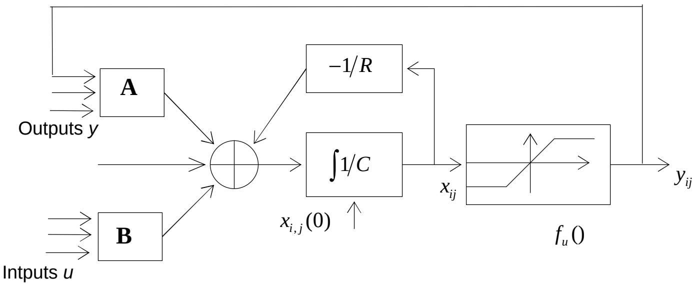


Fig. 5. The Cellular Neural network proposed in [36].

In [38][39] CNNs perform fast and distributed operation on frame images. The mathematical formulation for the network used is the following

$$\begin{aligned}\dot{x}_{ij}(t) = x_{ij}(t) + \sum_{k,l} A_{i,j;k,l} y_{kl} + \sum_{k,l} B_{i,j;k,l} u_{kl} + \\ + \sum_{k,l} \hat{A}_{i,j;k,l}(y_{kl}) + \sum_{k,l} \hat{B}_{i,j;k,l}(u_{kl}) + I_{ij}\end{aligned} \tag{1.3}$$

The motion compensation proposed aims at determining, inside the frame I_{n+k} , what are the object belonging to the frame I_n. Considering the frame $n+k$, the objects position in the previous frame n are computed by moving each object of the frame n in a $p\ x\ q$ - pixels window and comparing the result with the frame I_{n+k} .

The motion research is performed following a spiral trajectory. All the processing operation need to perform this research are made by the CNN fixing some values to the network parameters such as **A**, **B**, $\hat{\mathbf{A}}$, $\hat{\mathbf{B}}$, **x**, **I**, **u**, **y**.

Neuro fuzzy segmentation of Human image sequences

The modern video coding techniques in order to achieve better compression ratios allows different compressing methods applied to different objects of a the same video stream (object-based compression).

The advantage of using different compressions for different objects are strictly bounded to the capability of identify and extract the objects from the background of a video stream. Classical tools related to the generation of the region-based representations are discussed in [44] in which are reviewed state of art on this framework.

In [41] spatial and temporal information are combined to perform a neuro-fuzzy video segmentation of a videoconference streams (one person and a background). The approach consists of three main steps:

- clustering
- detection
- refinement

In the first step a fuzzy self-clustering algorithm is used to group similar pixels in the base frame of video stream, into fuzzy cluster. In detail frame image is divided into 4 x 4 pixels blocks, then block are grouped in segments by the clustering algorithm, these segments are then combined together in order to form larger clusters. Each cluster is represented by Gaussian membership functions (one for the luminance and one for each chrominance) with a given mean value and variance.

After fuzzy clustering is completed, the detection step starts. This second step detects human face and body and extracts them from background. Face segments are quite easy to be identified because they are characterized by values of chrominances within a reduced range and values of luminance with consistent variations.

Once the face area was identified the rest of body is assumed to lay in the area under the face, so possible body segments belong to that area. On the base of such analysis the clusters can be divided into:

- foreground
- background
- ambiguous region

A fuzzy neural network is constructed and trained in order to identify the ambiguous region too. The architecture of such neural network is represented in figure 6.

For pixels of each cluster the input of the network x_1, x_2, x_3 are the values of luminance and two crominances, with such inputs the output of the network will be 1 if the cluster (or only the pixel) totally covers the human object and 0 otherwise. The network's layers considering the architecture of figure 6 are:

- *Layer 1*: The *input layer* contains three nodes each of that transmit directly its input to the next layer.

– *Layer 2*: The *fuzzification layer* contains *N* groups of three neurons (*N* is the number of the fuzzy clusters). The output is computed as a Gaussian function: $o_{ij}^{(2)} = \exp\left[-\left(\frac{o_i^{(1)} - m_{ij}}{s_{ij}}\right)^2\right]$ m_{ij}, and s_{ij} are free parameter of the learning.

– *Layer 3*: The *inference layer* contains *N* neurons; the output of each neuron is $o_j^{(3)} = \prod_{i=1}^{3} o_{ij}^{(2)}$.

– *Layer 4*: The *output layer* contains only one neuron that perform the centroid defuzzification; it can be expressed as: $o^{(4)} = \frac{\sum_{j=1}^{N} c_j o_j^{(3)}}{\sum_{j=1}^{N} o_j^{(3)}}$

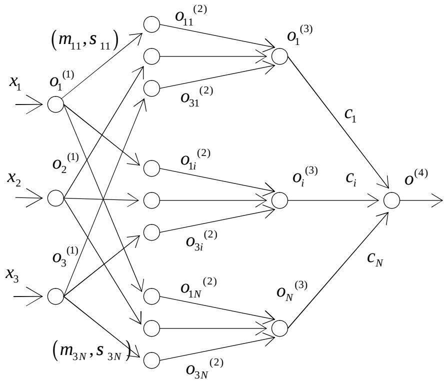


Fig. 6. The fuzzy neural network that perform the human objects refinement.

The free parameters (m_{ij}, s_{ij}, c_i) of the network are trained from foreground and background blocks, the training algorithm is a combination of SVD based least square estimator and gradient descendent method (hybrid learning).

Other approach on segmentation by fuzzy neural network are based on the fuzzy clustering of more complex data structure; data considered to perform the segmentation are together inter-frame information such as colour, shape, texture and contour, and intra-frame information such as motion information and object temporal shape.

In [42] good segmentation results are obtained by a two steps decomposition. The first step performs the image splitting in subsets, using an unsupervised neural network; the frame image is than divided into its clusters.

The hierarchical clustering phase reduces the complexity of the object structure then a final processing based on PCA (eigendecomposition) performs the final refinement and provides the final foreground-background segmentation.

Other approaches are based on a sub space representation of the video sequence[43]. The algorithm describes video sequences by the minimum set of maximally distant frames, selected on the base of semantic content, that are still able to describe the video sequence (Key Frames); these frames are collected in a Codebook. The core of the coding system is the Video Key Frames Codebook definition; it is based on video analysis in the vector space. The creation is performed by an unsupervised neural network, it consists into the storyboarding of the recorded sequence.

Image feature vectors are used to represent the images into the vector space; clustering all the images in feature vector space selects the smaller set of Video Key Frames used for defining the VKC.

Neural video compression: the image coding approach

The following sections detail two waveform video compression algorithms; the techniques proposed are based on feed forward and locally recurrent neural networks. The generalization of a still image compression approach inspired the technique of the first section [75]. In this context compression is achieved by finding some transform able to code images with a reduced number of parameters still representing the original image with a satisfying quality level; this technique is well known as *transform coding* [51].

Given the set of coefficient from of a portion of an image or a video frame, transform coding produces a reduced set of coefficients such that the reconstruction has the minimum possible distortion. This reduction is possible because most of the starting block energy is grouped in a reduced number of coefficients that became representative for the whole block.

The optimum transform coder, in the sense of mean square error, is the one that, for a fixed quantization, minimize the mean-square distortion of the reconstructed data; Karhunen-Loève transform respect this constraint.

In the framework of video compression this still image compression technique is applied jointly with a time decomposition therefore important issues are the space-time decomposition and the information compression.

Next sessions are dedicated to the image-video preprocessing and to some way to realize the neural transform coder.

Video frames pre-processing

Everyone can see in images uniform color areas, with a poor informative content, and areas with higher detail levels yielding much more information. Therefore using different compression ratios on areas with different activity levels, should provides a better quality on detailed areas, and higher compression ratios on areas with more uniform values.

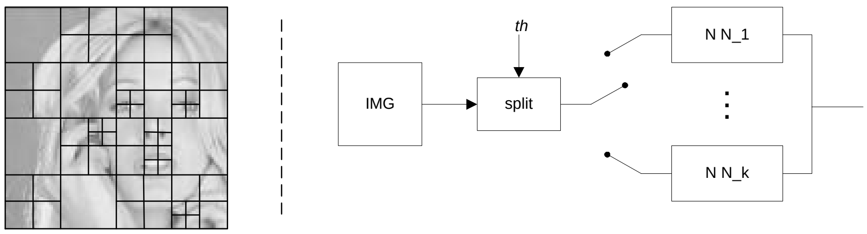
Therefore frame images are decomposed in sub blocks that are processed instead of processing the whole image. Such blocks can be divided in subclasses and coded with different coders to improve the performance of the compression [14][16][15].

In several papers block activity leads the coder to perform more or less compression; the idea consists in using different compression ratios on areas with different activity levels, in order to obtain a good quality on blocks with many details, and high compression ratios on the blocks with uniform values.

Suitable results can be reached dividing higher activity blocks on the base of theirs orientation: horizontal, vertical, diagonal. The best performance are obtained with the classification proposed in [15] in which such blocks are grouped according to nine possible orientations: two horizontal (one darker on the left, one darker on the right), two vertical, four diagonal and the last shaded.

Figure 7 shows a picture splitted in different size blocks by means of a quad-tree approach, based on the pixels variance measure: the bigger is the dimension of the block, the lower is the content detail, and vice versa.

Blocks with the same mask size are characterized by quite the same information amount and are grouped in order to perform a processing with the same neural network. Such Neural Network requires, in learning phase, training sets specific for the group. In this way each Neural Network is specialized for treating a particular class of sub-images, all characterized by quite the same activity.



a) b)

Fig. 7. a) Quad-tree segmentation; b) Adaptive size mask splitting block.

In Video sequences is possible to identify not only areas in which the details are more or less clear, but also areas in which there can be more action or not, or rather in which the subject is moving or not.

Identifying into the original sequence, sub-sequences in which the scene has reduced action, it is possible to assemble together similar frames and use the same quad-tree segmentation.

So useful video representation is obtained identifying adjacent frames with reduced dissimilarities, named *GOF* (Group Of Frames). Each *GOF* collects a *DA* (*Depth Activity*) number of frames grouped on the base of a threshold of similarity *th*; such frames have the same quad-tree segmentation structure.

The threshold is a critical issue in determining the *Depth Activity* because the number of frames collected into the same *GOF* strictly depend by *th* values.

Large values of threshold perform lower quality video restoring because frames are not represented by their own quad-tree structure; on the other hand, too small values of threshold perform a better video restoring quality but larger bit rate.

Groups of Frames identification, is based on the comparison of the prearranged threshold *th* with the variance between pixels of several close frames. It follows the *GOFs* generation algorithm:

1. given the first frame (keyframe) representing the reference image of GOF-i;
2. some frame n belong to the set GOF-i if: the variance of the image obtained by the difference between frame n and keyframe is under the threshold *th*.
3. finally the number of frames for which is valid step 2) gives the *DA*. Then the first frame of *GOF-i* is the keyframe, each following frame can be replaced by the differences between such frame and the keyframe; Figure 8 summarizes the generating process of the *GOF*.

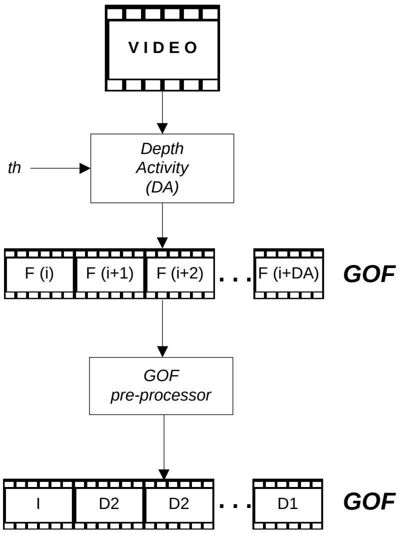


Fig. 8. Video processing that perform the Group Of Frames generation.

The images contained in every GOF are coded by a set of trained Neural Structures that will be detailed in dedicated sections. The frame *I* and the last one inside the GOF, frame *D1*, will be coded with a fitted Quad-tree structure as can be seen in figure 9. For each sub-block of the keyframe *I* and of the frame *D1* (first and last frames of the GOF) have to be transmitted, in addition to the compressed data content, information on the quad tree segmentation and network to be used for the specific sub-blocks, on the coding of the sub-block mean value, about the quantization and concerning the number of frames internal to the GOF.

Sub blocks of *D2* (the residual frame of the GOF) only require the information about the compressed data because they have the same segmentation of *D1*.

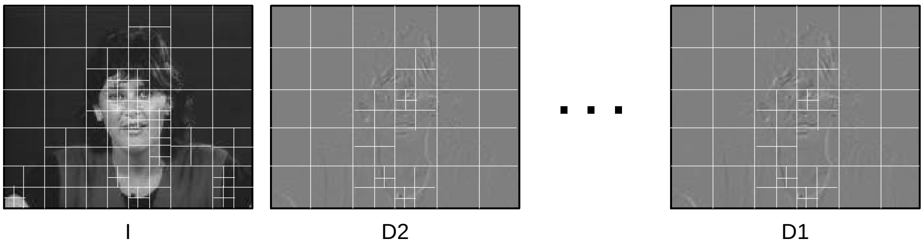


Fig. 9. QT schemes applied to the pictures within the GOF

The advantage in using the D2 frames resides in fact that, frames near to D1 frame, with high confidence level, have the same quad-tree segmentation structure as in figure 9; moreover such images are constituted for large parts of wide uniform color areas, therefore the mask applied to them will be principally constituted by large size blocks (e.g. 16x16) going to reduce the value of the bit-rate.

Figure 10 presents a blocks scheme of the processing from the video sequence to the compressed one.

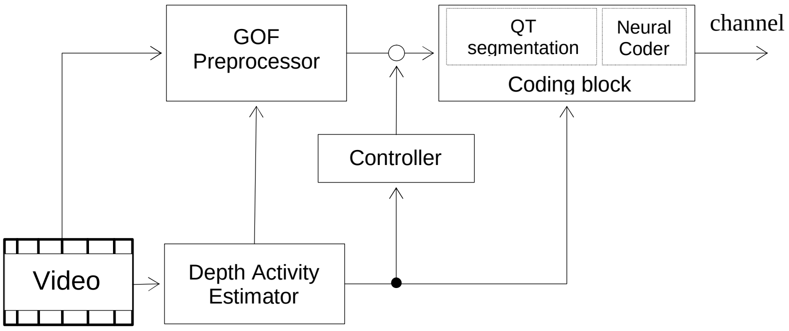


Fig. 10. Blocks scheme for the neural quad-tree video coding

The Video preprocessor, given the original video stream, establishes the value of the Depth Activity (DA); the GOF preprocessor performs the differences between frames ; the Controller selects keyframes and frames D1 and D2 which have to be segmented in different ways; the coding block after segmenting each frame into blocks, performs neural coding for each group of input block.

Neural feed-forward Compressor

Once organized the visual information in a collection of segmented images belonging to a GOF next step is to compress each image block.

In transform coding framework Karhunen - Loève transform represents signals on the basis of its principal component; considering only a reduced set of principal components is possible to decode the original video allowing reconstruction error (loose of visual quality) depending by the variance of the eigenvalues of the discarded eigenvector.

In detail, given an N-dimensional vector signal $\mathbf{x}$ ($n \times n$ pixels image), Karhunen - Loève transform represents it in the orthogonal space of the eigenvector of its autocovariance matrix (squared $N \times N$ matrix); assuming $\mathbf{W}$ as the $N \times N$ change basis matrix, no compression is performed [21]:

$$\mathbf{y} = \mathbf{W}\mathbf{x} \tag{1.4}$$

The vector $\hat{\mathbf{y}}$ represents the projection of the vector signal $\mathbf{x}$ on a subspace spanned by reduced number of eigenvectors ($m < N$).The representation error is not larger than the sum of the squared eigenvalues corresponding to the not chosen eigenvectors. Considering a change basis matrix $\mathbf{W}$ with rows ordered in such a way to minimize reconstruction error the result is a vector $\hat{\mathbf{y}}$, provides the better approximation of $\mathbf{x}$ given the subspace dimension [1].

$$\hat{\mathbf{y}} = \hat{\mathbf{W}}\mathbf{x} = \sum_{i=1}^{M<N} w_i x \tag{1.5}$$

KLT represents original vectors with reduced dimension so performs compression; moreover output vector coefficients are uncorrelated and therefore the redundancy due to the high degree of correlation between the neighbouring pixel is removed.

Application of KLT to video compression is not efficient because it depend on second order statistics (autocovariance matrix); moreover eigen-decomposition requires big computational effort considering the vector size in image framework. The statistical approach to KLT is not adequate to image coding application. Discrete Cosine Transform (performed via FFT) is able, into the image compression framework, to approximate quite well the KLT [51][45].

This reason inspire Neural approach on KLT, faster and computationally less intensive, to perform the solution of such a problem.

Linear Network: the Hebbian learning

Linear PCA is a solution to the problem of eigendecomposition of the autocovariance matrix. In [2] it was proposed a mechanism inspired to neurobiology, synaptic connections between neurons are modified by the learning; Hebb’s assumption is that if two neurons are both active at the same time the synaptic bound between them becmes stronger. In other word when input and output neurons have at the same time an high output value, the connection between them is reinforced (grow in value). The artificial neuron designed to perform the principal component extraction using Hebbian learning is the one in fig 11.

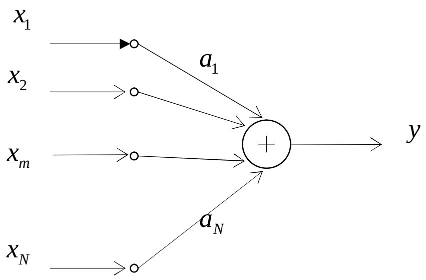


Fig. 11. Hebbian Neuron

Figure 11 simplifies the biological interaction each of one can be mathematically modelled as follows:

$$y = \mathbf{ax} \tag{1.6}$$

The Hebbian learning rule applied on this structure is:

$$\mathbf{a}[n+1] = \frac{\mathbf{a}[n] + m\mathbf{a}[n]\mathbf{x}[n]}{\left\|\mathbf{a}[n] + m\mathbf{a}[n]\mathbf{x}[n]\right\|} \tag{1.7}$$

In which m is the learning rate, $\|\bullet\|$ is the Euclidean norm. This rule has been shown to converge to the first principal component.

Generalizing the Hebbian learning is possible to find the first M principal component. The second principal component can be obtained taking again the first principal component of the set of data obtained by removing the first principal component from the original data and so on.

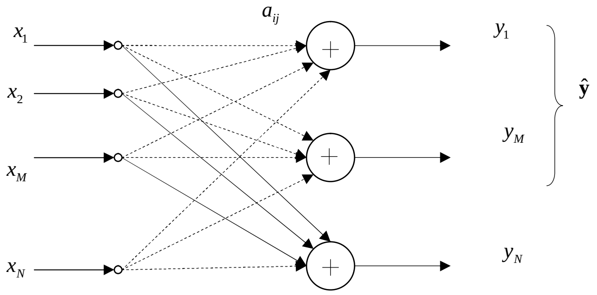


Fig. 12. Linear Network for Pricipal component extraction

The generalized Hebbian Algorithm, inside the learning, include also the orthogonalisation as shown in the following equation:

$$\mathbf{A}[n+1] = \mathbf{A}[n] + m\left[\mathbf{yx}^T - LT\left[\mathbf{yy}^T\right]\mathbf{A}[n]\right] \tag{1.8}$$

where the LT is the operator that set to zero all the element above the diagonal.
The generalized Hebbian learning provide the matrix **A** with the first M principal directions. An alternative approach in providing the first M principal component

was given by the APEX (Adaptive Principal component Extraction) network in which the so called Hebbian synapses are used together with the anti-hebbian one.

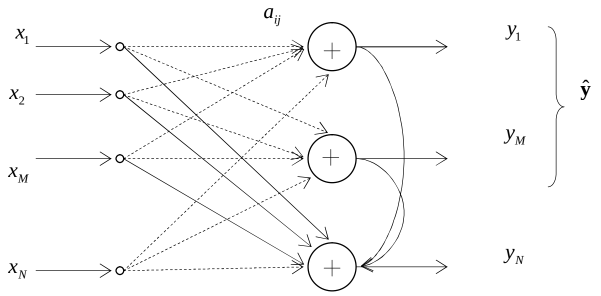


Fig. 13. The APEX network

Also his architecture has a biological justification. The m-th principal component can be computed on the base of the previous m-1; the c component are called anti Hebbian.

An exhaustive explanation about the Hebbian learning and about the algorithms inspired to its come out of the score of this section so for the learning rule of APEX network see[74].

Non linear Neural Network: MLP

In 1988, Cottrel, Murno and Zipper try to resolve the PCA problem with a two layer perceptron [5] trained with the so called Autoassociative Back Propagation; this work opened the way to a large number of future developments.
Figure 14 shows the proposed architecture. The first algorithm by Cottrel was referred to a linear structure; there each neuron is expressed as follows:

$$\hat{y}_i = \mathbf{a}^T{}_i\mathbf{x} \tag{1.9}$$

So:

$$\begin{aligned}\hat{\mathbf{y}} &= \mathbf{A}^T\mathbf{x}\\ \hat{\mathbf{x}} &= \mathbf{A}\hat{\mathbf{y}} = \mathbf{A}\mathbf{A}^T\mathbf{x}\end{aligned} \tag{1.10}$$

In this framework should be noted that with respect the (1.4) and (1.5) the (1.10) allow also different optimum solutions given by all **A**=**WQ**$^{-1}$ (in which **Q** is such that **QQ**$^{-1}$=**I**).

This underline the issue that neural networks can achieve the same compression ratio as KL transform without reaching this own eigenvector matrix.

Other approaches [4] develop neural networks with nonlinear sigmoidal function that can be trained during the learning. This approach reaches better results with respect to the linear network [3].

One of the most critical issue of the application of neural PCA is the fixed compression ratio of each processed block: the network performs the compression with a quite low distortion level uniform blocks but with higher distortion the less uniform one.

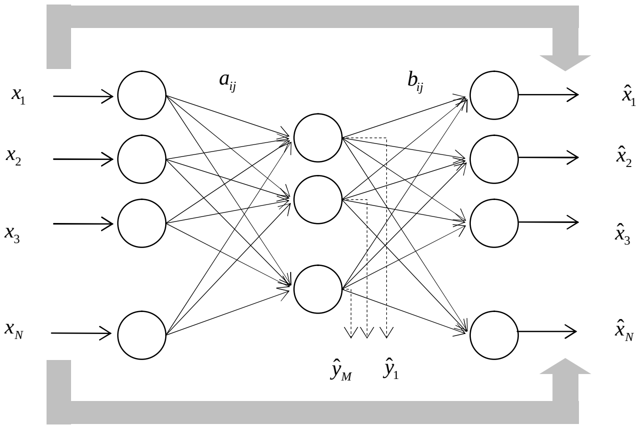


Fig. 14. Multi Layer perceptron performed for Autoassociative Back Propagation.

To overcome this problem Size Adaptive Networks [6] uses different trained networks to perform a compression which strongly depend by the block activity. This allows higher compression of blocks with a low activity level but and a good detail recovery of the block with an higher one.

As is introduced in the dedicated section the a quad-tree algorithm segment images into several dimension block, on the base of activity level. By this way images are segmented in blocks of 4 x 4, 8 x 8 and 16 x 16 as shown in figure 15.

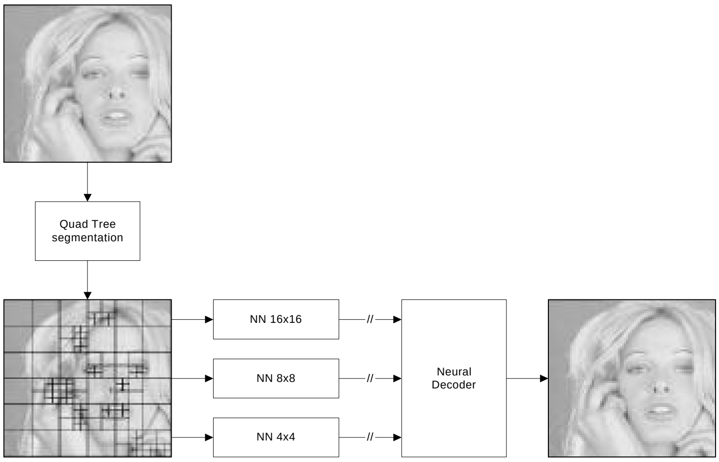


Fig. 15. Adaptive size mask visual information compression

Three kinds of neural structures are developed to process that different size blocks: each network has an hidden layer of 8 neurons but into the input and output layers has so mach neuron as pixels are into the block. The output of each neuron is quantized with 4 bit.

It should be noted that learning is an important issue in this kind of application. In order to improve the learning capability advances in adapting sigmoidal function has been developed; in neural networks spline adaptive models are used instead of fixed sigmoidal functions [8].

Performance of the video compression are usually valuated on the base of Peack Signal to Noise Ratio (1.11) calculated as a the MSE in dB ad bit rate (in kbit/sec).

$$PSNR = 10 \cdot \log_{10}\left(\frac{256^2}{\frac{1}{M \times N} \cdot \sum_{m=1}^{M} \sum_{n=1}^{N} \left[pix_{org}(m,n) - pix_{comp}(m,n) \right]^2} \right) \tag{1.11}$$

Figure 16 shows several PSNR values obtained compressing Missa benchmark by processing GOFs with different threshold levels.

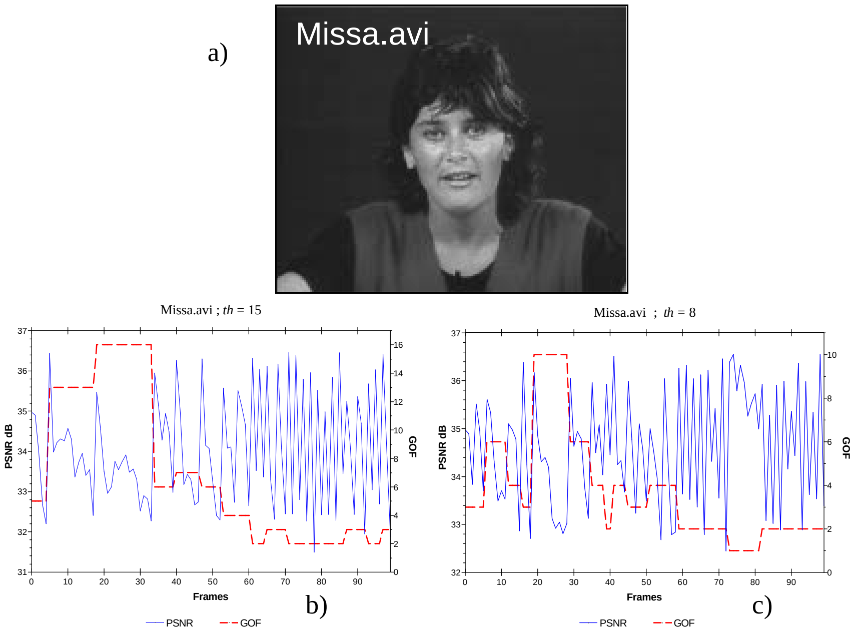


Fig. 16. a) Missa avi movie segmented and compressed; b-c) show PSNR considering the GOF evolution with two different level of threshold.

Table 1 shows several PSNR and Br values of the video Missa at the threshold changing. From table 1 it is easy to see that considering a relaxed value of threshold (an higher value) will produce a gain in compression level (bit rate) but diminish the quality of the recovered video.

Table 1. Peack Signal to Noise Ratios (PSNR) and bit error Rate (Br) on changing threshold of the Missa video.

	th = 8		th = 15		th = 30	
	PSNR (dB)	Br (kbps)	PSNR (dB)	Br (kbps)	PSNR (dB)	Br (kbps)
Missa	34,62	205,63	34,02	166,05	33,02	152,85
Susi	31,11	469,53	30,91	422,31	30,38	361,52

Hierarchical Linear Structure

Two layers Non linear Neural Network just widely exposed reach some suitable performance in video compression: it derives its success from several advantage such as short time encoding decoding no explicit use of codebooks.

However it consider only the correlation between pixel within the same segmented block so only a limited level of compression can be obtained. On a theoretical point of view a better decorrelation level (and then compression) can be reached considering, as input of a neural structure, not only the block itself but also nearer one.

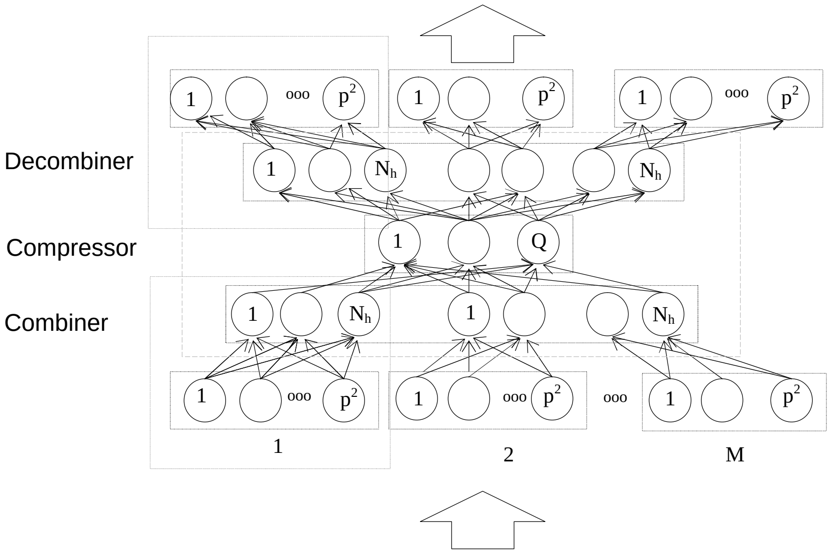


Fig. 17. Multilayer network for high order data compression-decompression.

This suitable performance is obtained by Hierarchical Neural Networks applications [7]. The idea is to divide a the scene or part of it into *N* disjoint sub-scenes each of one is segmented in $n \times n$ pixels blocks, the group of blocks are processed together by the hierarchical structure represented in figure 17.

The Hierarchical Neural Network is not fully connected; it consist of input, hidden and output layers.

While the input and the output layers are single layer composed by N input blocks (one for each section of the image) where each block has n^2 neurons; the hidden-layer section consist of tree layers: combiner, compressor and decombiner layer. The combiner level is not fully connected with the input one.

Although the learning of this structure could be performed by the classical back propagation the so called *Nested training algorithm* provides better performances. NTA is a three phases training, one for each part of the structure:

- *OLNN* (Outer loop neural network) performs the training of the fully connected network constructed by input layer, combiner layer and output layer; a standard back propagation is applied to the structure in which the desired output is equal to the input; the training set is given by images segmented blocks
- *ILNN* (Inner Loop Neural Network) performs the training of the hidden fully connected layers: Combiner, Compressor and Decompressor
- After the *OLNN* and the *ILNN* are trained, their weights are used to construct the overall network

It should be noted that this hierarchical structure perform inter block decorrelation in order to achieve a better compression level. The use of two layer perceptron (not hierarchical structure) with spline adaptive activation functions riches the same performance in term of image quality and compression level requiring a more simple structure.

Recurrent neural network

In video framework processing in a time space domain allows to perform prediction algorithms not only within the same frame but also with respect to near frame. Many widespread video coding techniques make use of some tricks in order to take advantage of such additional information, e.g. the motion compensation in MPEG.

In the field of the neural networks consistent advantages are reached including dynamic inside the structure of the Neural Network; the networks just introduced in previous section (see figure 14) are modified in order to follow the dynamic characteristics of the video sequences. This precaution allows either to improve the quality of the restored video, fixed the bit-rate, or further reduce the compression level [78]. Dynamic behavior in Multi Layer Perceptrons can be obtained by two different approaches:

- *Local approach*: introducing a dynamical (e.g. ARMA) model of the neuron
- *Non Local Approach*: introducing the feedback of the whole neural network

In both approaches, given an input at time lag *n* $x[n]$, it may influence a the output at the time lag *n-h* $y[n-h]$. In case of asymptotic stability, $y[n-h]/\partial x[n]$ this derivative goes to zero when *h* goes to infinity.

The value of *h* for which the derivative becomes negligible is called *temporal depth*, whereas the number of adaptable parameters divided by the temporal depth is called *temporal resolution*.

The architecture used in this context is the IIR-MLP proposed by Back-Tsoi [10][11] where static synapses are substituted by conventional IIR adaptive filters, as depicted in figure 18:

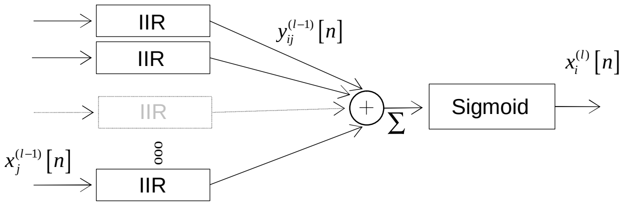


Fig. 18. Locally recurrent neuron for Multilayer Neural Network.

In literature there exists several algorithms to train such kind of networks, although a comprehensive framework is still missing.

In [9] it was introduced a very performing algorithm for the learning of the so called locally recurrent neural networks. It is a gradient rule based on the recursive back-propagation algorithm.

The learning of the locally recurrent neural network is performed by a new gradient-based on-line algorithm [9], called causal recursive back-propagation (CRBP); it presents some advantages with respect to the already known on-line training methods and the well known recursive back propagation. This CRBP algorithm includes the Back Propagation as particular cases [12][13].

Locally recurrent Neural Network is designed introducing an ARMA model on the site of linear synapses, figure 19 shows the structure of the network. The forward phase at time lag *n* is described by the following equations evaluated for the layers $l=1,...,M$ and the neurons $m=1,..,N_l$.

$$y_{km}^{(l)}[n]=\sum_{p=0}^{L_{km}^{(l)}-1} w_{km(p)}^{(l)} x_m^{(l-1)}[n-p]+\sum_{p=1}^{I_{km}^{(l)}} v_{km(p)}^{(l)} y_{km}^{(l)}[n-p] \qquad (1.12)$$

$$x_k^{(l)}[n]=sgm\left(\sum_{m=0}^{N_{l-1}} y_{km}^{(l)}[n]\right) \qquad (1.13)$$

Given $\Phi^{(l)}[n]$ the set of weights of layer *l* at the time lag *n*, the updating rule is:

$$\Phi^{(l)}[n+1]=\Phi^{(l)}[n]+\Delta\Phi^{(l)}[n+1-D_l] \qquad (1.14)$$

in which:

$$D_l = \begin{cases} 0 & \text{if } \; l = M \\ \sum_{i=l+1}^{M} \max_{n,m}\left(L_{nm}^{(i)} - 1\right) & \text{if } \; 1 \le l \le M \end{cases} \tag{1.15}$$

With $\left(L_{nm}^{(i)} - 1\right)$ order of the moving average part of the synapse of the *n*-th neuron of the *l*-th layer relative to the *m*-th output of the (*l*-1)-th layer.

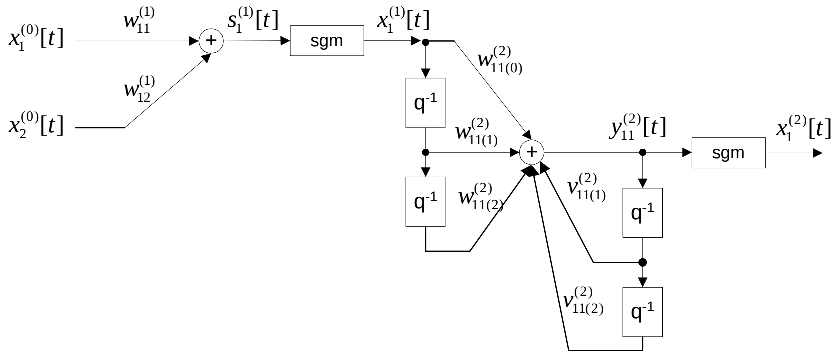


Fig. 19. Locally recurrent ARMA model for a Multialyer Perceprton.

The Causal Recursive Back Propagation learning rules are:

$$\Delta\Phi_{km(p)}^{(l)}[n+1] = m e_k^{(l)}[n] s\dot{g}m\left[s_k^{(l)}[n]\right]\frac{\partial s_k^{(l)}[n]}{\partial \Phi_{km(p)}^{(l)}} \tag{1.16}$$

$$e_k^{(l)}[n] = \begin{cases} e_k[n] & l = M \\ \sum_{q=1}^{N_{l+1}} \sum_{p=0}^{Q_{l+1}} d_q^{(l+1)}[n+p] \frac{\partial y_{qk}^{(l+1)}[n+p]}{\partial x_k^{(l)}[n]} & l = (M-1),\ldots,1 \end{cases} \tag{1.17}$$

The CRBP algorithm is computationally simple and the application to the video compression produce some suitable performances.

The proposed architecture is applied as Neural coder in the Coding Block referring to the architecture shown in figure 10; so according to the general architecture, this neural coder compresses, receive image block from quad tree segmentation.

Learning of a locally recurrent neural network for the video compression, is a delicate issue due to fact that recurrent networks are sensitive to a large number of factors, as for instance the type of videos content in the training set, or the video length, or the way in which the examples are presented.

Such sensitivity can compromise the correct learning of the network, altering so the end results e.g. it could produce artifacts on the restored video. Most common artifacts are the so called "*regularities*" and the so called "*memory effect*", both of them can be avoided by a special care into training and designing the structure.

An example of "*regularities*" is shown in figure 20; it can be avoided by reducing the length of the video training set.

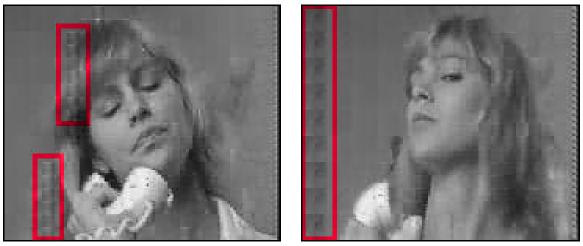


Fig. 20. Regularities effects (into the boxes) in two frames of a video sequences.

In phase of video restoring, it can happen that, especially on uniform color backgrounds, the image of objects not more present in the scene remains impressed, because of the presence of the delay lines in the synapse, this is the so called "*memory effect*" (see it is shown in figure 21). This artifact can be avoided by a special care in dimensioning the neurons dynamics and than the number of tap of the ARMA filter.

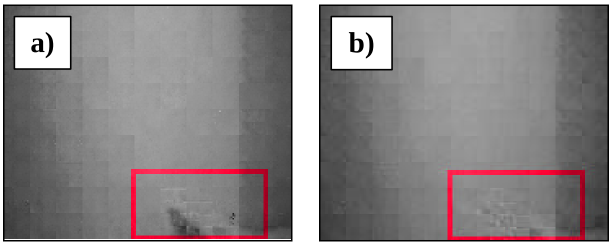


Fig. 21. The memory effect in two frames of a video sequences

In this context a lot of advantage can be found by using locally recurrent neurons only in the second layer of the structure of figure 14.

It could be observed, from the examples of figures 20-21 too, that most of artifacts are in the "background" of the scene.

Recurrent Neural Networks reach a good learning level in more dynamic part of the scene but not in the static background sections.

In order to overcome this problem after the segmentation of the scene could be used an hybrid approach to perform the training of the network.

Since static neural networks perform better more static subscenes this can be used to perform the compression of the 16×16 blocks, the one with the lowest activity; recurrent neural network can be used to code 4×4 and 8×8 block with an high detail level.

This approach achieve a different processing for the lower and the higher activity blocks, not only for the network size but also in structure and learning. Performance obtained with hybrid approach are collected in the table 2.

	Susi_02	Susi_03	Susi_04

	6 Neuron hidden Laye	5 Neuron hidden Laye	4 Neuron hidden Laye
BR (kbs)	433,38	372,51	319,32
PSNR (dB)	28,92	28,45	28,01

Table 2. Mean value of bit rate and peak signal to noise ratio reached with three different kind of neural network all of them are IIR MLP set with the dynamic synapse.

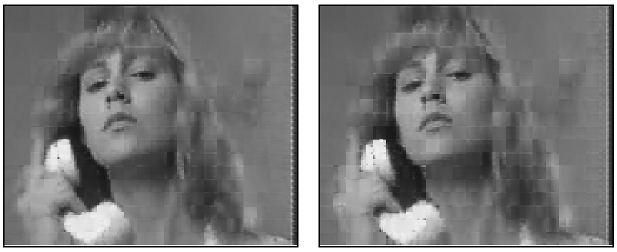


Fig. 22. Frames of the Suzi video compressed and recovered with Suzi_02 (showing no block effect) network and Suzi_04 (showing block effect).

However the improvement of recurrent neural network with respect to results obtained with a static network are not so evident in comparing results of table 2 with the one collected in table 1.

The evidence on that improvement should be observed in seeing video sequence: more fluid movements are performed.

References

[1] Jiang J (1999) Image compression with neural network – A survey. In: Signal Processing and image communications, vol 14, 1999, pp 737-760.

[2] Hebb D O(1949) The organizazion of behaviour. New York, Wiley, 1949

[3] Dony R D, Hykin S (1995) Neural network approach to image compression. Proc. IEEE 83, vol 2, February 1995, pp 288-303.

[4] Kohno R, Arai M, Imai H (1990), Image compression using neural network with learning capability of variable function of a neural unit. In: SPIE vol 1360, Visual Communication and Image processing ‘90, pp 69-75, 1990.

[5] Cottrel G W, Munro P, Zipser D (1988), Image Compression by back propagation and examples of extensional programming. In: Sharkey. N. E. (Ed.) Advances in cognition science (Ablex norwood, NJ 1988).

[6] Parodi G, Passaggio F (1994), Size-Adaptive Neural Network for Image Compression. International Conference on Image Processing, ICIP '94, Austin, TX, USA.

[7] Namphon A, Chin S H, Azrozullah M (1996), Image compression with a Hierarchical Neural Network, IEEE Transaction on Aereospace and electronic System, vol 32, No.1 January 1996.

[8] Guarnirei S, Piazza F, Uncini A, (1999) Multilayer Feedforward Networks with Adaptive Spline Activation Function, IEEE Trans. On Neural Network, vol 10, No. 3, pp. 672-683.

[9] Campolucci P, Uncini A, Piazza F, Rao B D (1999), On-Line Learning Algorithms for Locally Recurrent Neural Networks. IEEE Trans. on Neural Network, vol 10, No. 2, pp 253-271 March 1999.

[10] Back A D, Tsoi A C (1991) FIR and IIR synapses, a new neural network architecture for time series modelling. Neural Computation, vol 3, pp. 375-385.

[11] Back A D, Tsoi A C (1994) Locally recurrent globally feedforward networks: a critical review of architectures, IEEE Trans. Neural Networks, vol 5, pp 229-239.

[12] Rumelhart D E, Ton G E, Williams R J, (1986) Learning internal representations by error propagation, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol 1, D. E. Rumelhart, J. L. McClelland, and the PDP Research Group, Eds. Cambridge, MA: MIT Press.

[13] Widrow B, Lehr M A, (Sept 1990) 30 years of adaptive neural networks: perceptron, madaline and backpropagation, Proc. IEEE, vol 78, pp 1415-1442.

[14] Cramer C (1998) Neural Network for image and video compression: A review. European Journal of Operational research pp 266-282.

[15] Marsi S, Ramponi G, Sicuranza L (1991) Improved neural structure for image compression. In: Proceedings of the international conference on acoustic speech and signal processing Toronto, Ont., IEEE Piscataway, NJ, 1991 pp.2821-2824.

[16] Zheng Z, Nakajiama M, Agui T (1992) Study on image data compression by using neural network. In: Visual communication and image processing'92, SPIE 1992, pp 1425-1433.

[17] Gray R M (1984) Vector quantization. In: IEEE Acoustic and Speech Signal Processing. Apr. 1984, pp 4-29.

[18] Goldberg M, Boucher P R, Shliner S (1988) Image compression using adaptive vector quantization. In: IEEE Trans. Communication, vol 36, 1988, pp 957-971.

[19] Nasrabadi N M, King R A (1988) Image coding using vector quantization: A review. In: IEEE Transaction on communication, vol 36, 1988, pp 957-971.

[20] Nasrabadi N M, Feng Y (1988) Vector quantization of image based upon Kohonen self organizating features map. In: IEEE Proceeding of international conference of Neural Networks, S.Diego, CA, 1988, pp.101-108.

[21] Haykin S (1998) Neural Networks: A Comprehensive Foundation. In: Prentice Hall, 06 July, 1998

[22] Kohonen T (1990) The self organizing map. In: Proc. IEEE, vol 78, pp. 1464-1480, Sept 1990.

[23] Poggi G, Sasso E (1993) Codebook ordering technique for address predictive VQ. In: Proc. IEEE Int. Conf. Acoustic and Speech and Signal Processing '93, pp. V 586-589, Minneapolis, MN Apr. 1993.

[24]Liu H, Yum D J J (1993) Self organizing finite state vector quantization for image coding. In: Proc. of international Workshop on Application of neural networks in telecommunications, Hillsdale, NJ: Lawrence Erlbrume Assoc., 1993.

[25] Forster J, Gray R M, Dunham M O (1985) Finite state vector quantization of waveform coding. In: IEEE transaction on information Theory, vol 31, 1985, pp 348-359.

[26] Luttrel S P(1989) Hierarchical vector quantization. In : IEE Proc. (London), vol 136 (Part I), pp 405-413, 1989

[27] Li J, Manicopulos C N (1989) Multi stage vector quantization based on self organizing feature map. In: SPIE vol 1199, visual Communic and Image Processing IV (1989), pp. 1046-1055.

[28] Weingessel A, Bishof H, Jornik K, Leish F (1997) Adaptive Combination of PCA and VQ neural networks. In: Letters on IEEE Transaction on Neural Network, vol.8 no. 5, Sept 1997.

[29] Huang Y L, Chang R F (2002) A new Side-Match Finite State Vetor Quantization Using Neural Network for image coding. In: Journal of visual Communication and image reppresentation vol 13, pp 335-347.

[30]Noel S, Szu H, Tzeng N F, Chu C H H, Tanchatchawal S (1999) Video Compression with Embedded Wavelet Coding and Singularity Maps. In: 13th Annual International Symposium on Aerospace/Defense Sensing, Simulation, and Controls, Orlando, Florida, April 1999.

[31] Szu H, Wang H, Chanyagorn P (2000) Human visual system singularity map analyses. In: Proc. of SPIE: Wavelet Applications VII, vol 4056, pp 525-538, Apr. 26-28, 2000.

[32] Hsu C, Szu H (May 2002) Video Compression by Means of Singularity Maps of Human Vision System. In: Proceedings of World Congress of Computational Intelligence, May 2002, Hawaii, USA.

[33] Buccigrossi R, Simoncelli E (Dec. 1999) Image Compression via Joint Statistical Characterization in the Wavelet Domain. In: IEEE Trans. Image Processing, vol 8, no 12, pp 1688-1700, Dec. 1999.

[34] Shapiro J M (1993) Embedded Image Coding Using Zerotrees of Wavelet Coefficients. In: IEEE Trans. Signal Processing, vol. 41, no. 12, pp 3445-3462, Dec. 1993.

[35] Skrzypkowiak S S, Jain V K (2001) Hierarchical video motion estimation using a neural network. In: Proceedings, Second International Workshop on Digital and Computational Video 2001, 8-9 Feb. 2001 pp 202-208.

[36] Milanova M G, Campilho A C, Correia M V (2000) Cellular neural networks for motion estimation. In: International Conference on Pattern Recognition, Barcelona, Spain, Sept 3-7, 2000. pp 827-830.

[37] Toffels A, Roska A, Chua L O (1996) An object-oriented approach to video coding via the CNN Universal Machine. In: Fourth IEEE International Workshop on Cellular Neural Networks and their Applications, 1996, CNNA-96, 24-26 June 1996, pp 13-18.

[38] Grassi G, Greco L A (2002) Object-oriented image analysis via analogical CNN algorithms - part I: Motion estimation. In: 7th IEEE International Workshop Frankfurt, Germany 22 - 24 July 2002.

[39] Grassi G, Grieco L A (2003) Object-oriented image analysis using the CNN universal machine: new analogic CNN algorithms for motion compensation, image synthesis, and consistency observation. In: IEEE Transactions on Circuits and Systems I, vol 50, no 4 , April 2003, pp 488 – 499.

[40] Luthon F, Dragomirescu D (1999) A cellular analog network for MRF-based video motion detection. In: IEEE Transactions on Circuits and Systems, vol 46, no 2, Feb 1999 pp 281-293.

[41] Lee S J, Ouyang C S, Du S H (2003) A neuro-fuzzy approach for segmentation of human objects in image sequences. In: IEEE Transactions on Systems, Man and Cybernetics, Part B vol 33, no3, pp 420-437.

[42] Acciani G, Guaragnella C (2002) Unsupervised NN approach and PCA for background-foreground video segmentation. In: Proc. ISCAS 2002, 26-29 May 2002, Scottsdale, Arizona, USA

[43] Acciani G, Girimonte D, Guaragnella C (2002) Extension of the forward-backward motion compensation scheme for MPEG coded sequences: a sub-space approach. In: 14th International Conference on Digital Signal Processing, 2002. DSP 2002 vol 1, 1-3 July 2002 pp 191 - 194.

[44] Salembier P, Marqués F (1999) Region-based representations of image and video: Segmentation tools for multimedia services. In: IEEE Trans. on Circuits and Systems for Video Technology, vol 9, no 8, pp 1147-1169, December 1999.

[45] Ebrahimi T, Kunt M (1988) Visual data compression for multimedia applications. In: Proceedings of the IEEE, vol 86, no 6, June 1998, pp 1109- 1125.

[46] The International telegraph and telephone Consultative Committee (CCITT) (1992) Information technology - Digital Compression and coding of continuous – tone Still Image Requirements and guidelines. Rec T.81, 1992.

[47] Pennebaker W, Mitchell J (1992) JPEG Still Image Data Compression Standard. Van Nostrand Reinhold, USA, 1992.

[48] Christopoulos C, Skodras A, Ebrahimi T (2000) The JPEG2000 still image coding system: an overview. In: IEEE Transactions on Consumer Electronics, vol 46, no. 4, pp. 1103-1127, November 2000.

[49] ISO/IEC FDIS15444-1:2000 Information Technology – JPEG 2000 Image Coding System. Aug. 2000.

[50] ISO/IEC FCD15444-2:2000 Information Technology – JPEG 2000 Image Coding System: Extensions. Dec. 2000.

[51] Egger O, Fleury P, Ebrahimi T, Kunt M (1999) High-Performance Compression of Visual Information-A Tutorial Review-Part I: Still Pictures. In: Proceedings of the IEEE, vol. 87, no 6, June 1999.

[52] Torres L, Delp E (2000) New Trends in Image and Video Compression. In: EUSIPCO '2000: 10th European Signal Processing Conference, 5-8 September, Tampere, Finland,2000.

[53] CCITT SG 15, COM 15 R-16E (1993), ITU-T Recommendation H.261 Video Codec for audiovisual services at p x 64 kbit/s. March 1993.

[54] Côtè G, Erol B, Gallant M, Kossentini F (1998) H.263+: Video Coding at Low Bit Rates. In: IEEE Transaction on Circuits and Systems for video technology, vol 8, no 7, Nov 1998.

[55] Côtè G, Winger L (2002) Recent Advances in Video Compression Standards. In: IEEE Canadian Review, Spring 2002.

[56] CCITT SG 15 ITU-T Recommendation H.263 Version 2 (1998) Video coding for low-bitrate communication. Geneve. 1998.

[57] Noll P (1997) MPEG digital audio coding. In: IEEE Signal processing Magazine vol 14, no 5, pp 59-81, Sept 1997.

[58] ISO/IEC 11172-2:1993 Information Technology (1993) Coding of moving pictures and Associated Audio for digital Storage media at up to 1.5 Mbits/s. Part 2.

[59] Sikora T (1997) MPEG Digital Video coding Standard. In: IEEE Signal Processing Magazine, vol 14, no 5, Sept 1997 pp.82-100.

[60] ISO/IEC 13818-2, Information Technology (2000) Generic coding of Moving Pictures and Associated Audio Information. Part 2.

[61] Haskell G B, Puri A, Netravali A N (1997) Digital video: an introduction to MPEG-2. Digital Multimedia, standard Series. In: Chapman & Hall 1997.

[62] ISO/IEC 14496-2:2001 Information Technology. Coding of audio-visual objects. Part 2.

[63] Grill B (1999) The MPEG-4 General Audio Coder. In: Proc. AES 17th International Conference, Set 1999.

[64] Scheirer E D (1998) The MPEG-4 structured audio Standard. In: IEEE Proc. On ICASSP, 1998.

[65] Koenen R (2002) Overview of the MPEG-4 Standard-(V.21-Jeju Version). ISO/IEC JTC1/SC29/WG11 N4668, March 2002.

[66] Aizawa K, T. S. Huang, Model Based Image Coding: Advanced Video Coding techniques for low bit-rate applications. In: Proc. IEEE, vol 83, no 2, Feb. 95.

[67] Avaro O, Salembier P (2001) MPEG-7 systems: Overview. In: IEEE Transaction on circuit and system for video Tecnology, vol 2. no 6, June 2001.

[68] ISO/IEC JTC1/SC29/WG11 N3933, Jan 2001. MPEG-7 Requiremens document.

[69] Manjunath B S, Salambier P, Sikora T (2002) Introduction to MPEG-7: multimedia content description language. In: Jhon Wiley & Sons 2002.

[70] Martínez J M, MPEG-7 Overview (version 9), ISO/IEC JTC1/SC29/WG11N5525, March 2003

[71]Burnett I, Walle R W, Hill K, Bormans J, Pereira F (2003) MPEG-21: Goals and Achievements. In: IEEE Computer Society, 2003

[72] Bormans J, Hill K (2002) MPEG-21 Overview v5, ISO/IEC JTC1/SC29/WG11 N5231, October 2002

[73] Saupe D, Hamzaoui R, Hartenstein H (1996) Fractal image compression: An introductory overview. In: Technical report, Institut für Informatik, University of Freiburg, 1996.

[74] Kung S Y, Diamantaras K I, Taur J S (1994) Adaptive Principal component extraction (APEX) and application. In: IEEE Trans. Signal. Processing vol 42 (May 1994) pp 1202-1217.

[75] Piazza F, Smerilli S, Uncini A, Griffo M, Zumino R, (1996) Fast Spline Neural Networks for Image Compression. In: WIRN-96, Proc. Of the 8th Italian Workshop on Neural Nets, Vietri sul Mare, Salerno, Italy.

[76] Skrzypkowiak S S, Jain V K (1997) Formative motion estimation using affinity cells neural network for application to MPEG-2. In: Proc. International Conference on Communications, pp 1649-1653, June 1997.

[77] ISO/IEC JTC1/SC29/WG11, ITU-T VCEG: working draft number 2 of Joint Video team standard”.

[78] Topi L, Parisi R, Uncini A (2002) Spline Recurrent Neural Networks for Quad-Tree Video Coding. In: WIRN-2002, Proc. Of the 13th Italian Workshop on Neural Nets, Vietri sul Mare, Salerno, Italy, 29-31 May 2002.

"Video coding for low bitrate communication," ITU-T SG XVI, DRAFT 13 H.263+ Q15 A-60 rev. 0, 1997.

Sicuranza G. L., Ramponi G., Marsi S., (1990) "Artificial Neural Networks for Image Compression", Electronics Letters, vol. 6, pp. 477-479.

Kiely, A. B. and M. Klimesh, "A New Entropy Coding Technique for Data Compression," IPN PR 42-146, April-June 2001, pp. 1-48, August 15, 2001.

[] M J Weinberger, J J Rissanen, R B Arps (1996), Application of universal context modellingto lossless compression of grey scale images, IEEE Trans. On Image processing. 5 (4) 1996 pp. 575-586.