Object-based Video compression using neural networks

Soumaya GHORBEL[1], Maher BEN JEMAA[2] and Mohamed CHTOUROU[1]

[1] Research unit of Intelligent Control, design & Optimization of complex Systems (ICOS)
University of Sfax, National Engineering School of Sfax, BP W 3038 Sfax – Tunisia

[2] REsearch, Development and Control of Distributed Applications (REDCAD)
University of Sfax, National Engineering School of Sfax, BP W 3038 Sfax – Tunisia

Abstract

This paper presents a n ew object-based video compression approach. It consists on pr edicting video objects motions throughout the scene. Neural networks are used to carry out the prediction step. A multi-step- ahead prediction is performed to predict the video objects trajectories over the sequence. In order to reduce video data, only the background of the video sequence is transmitted with the different detected video objects as well as their initial properties such as placement and dimensions. Experimental results show the effectiveness of the proposed approach in terms of the compression rates.

Keywords: *video compression, object tracking, neural network, multi-step-ahead prediction.*

1. Introduction

The video was integrated into all sectors of modern communication, even for low-bandwidth services like mobile communications. Thus, effective techniques for analysis, description and video compression are important areas and have great interests. Various standards for video compression exist. The ISO Moving Picture Experts Group (MPEG) concern towards compressed video storage, whereas the International Telecommunications Union (ITU) addresses real-time multi-point or point-to-point communications over a network.

Generally video sequences have strong correlations in the same frame and between successive frames. There are high redundancies in the same frame, the spatial redundancies, and between successive frames, the temporal redundancies. Most of video compression techniques aim to exploit these correlations and high redundancies in order to achieve better compression.

Video compression techniques can be categorized into four main axes [1]. There has been object-based technique which considers that a v ideo sequence is a collection of different objects [2]. Objects are usually extracted by a segmentation step [3]. Each object has its own shape, texture and motion representation and can be processed differently. Coding each object differently provides different compression ratio for each one and allows direct access and manipulation. MPEG-4 relies on this idea. Different distortion levels for different parts of the scene may be accepted and each object has its own stream. The second axis of video compression techniques is the waveform compression which uses the time as a t hird dimension. Such applications we find the DCT and the wavelets [4-5]. Many researches have been carried out based on this technique. In [6] for example, Ouni et al proposed a low complexity DCT based video compression method. This latter consists on 3D to 2D transformation of the video frames; that is means a temporal redundancy projection of each group of pictures into spatial domain. The result is combined with spatial redundancy in one representation and then is JPEG compressed. However, this approach is efficient only for low motion sequences and cannot be employed for fast motion sequences. The third axis of video compression techniques is the model based one which performs video analysis and structural 2D or 3D model synthesis [7]. The fractal based techniques represent the last axis of video compression techniques. In this framework, successful approaches which are applied to image coding are extended to video applications [8]. An overview on video compression standards can be found in [1, 9-10].

In this paper, we will focus on object-based video compression techniques in which motion estimation is performed. Block matching algorithm is one of the most used for motion estimation techniques. MPEG and H26x have used this approach for efficient encoding [11]. Joumana et al in [12], exploited the spatial prediction since adjacent blocks in a frame have similar motion activity. A model for Kalman filtering of motion information is generalized. Several frameworks are proposed in [13] relying on Kalman filter to exploit the intra-redundancy between motion vectors of a macroblocks group in a frame, as well as the inter-redundancy with macroblocks from a previous frame. Kalman filter is used to predict the position of a block in the next frame knowing its actual position. Significant improvements in video compression efficiency, as shown in [14], have been achieved by introducing intensive spatial-temporal prediction. In [15], the authors proposed a saliency-based attention prediction to detect the interesting regions in the video; only a small number of selected attention regions are encoded with high

priority to keep a high subjective quality, while less interesting regions are treating with low priority. Thus, prediction techniques are very important for video compression based on motion estimation.

In this work, we suggest to employ neural networks to achieve prediction task in video compression technique based on motion estimation. In fact, neural networks are known as well promising tools for prediction. Also, in the recent years, they have been successfully applied to video compression. They have been used for intra-frame coding, object segmentation, object clustering and motion estimation [16].

The remainder of this paper is organized as follows: an overview of video compression process based on motion estimation is presented in section2. Section 3 introduces the new neural network based video compression approach. It describes the neural network based prediction and how it is employed to accomplish video compression. The simulation results are presented in section 4 followed by a conclusion and future work in the section 5.

2. Video compression process

The entire compression decompression process in most of video coding based on motion estimation is basically the same as is illustrated in Fig. 1. These video coding use the hybrid coding approach; they exploit simultaneously the intra-frame and inter-frame redundancies to encode the video [17]. The intra-frame coding compresses a f rame independently of other frames. This frame, considered as a key frame noted by I-frame, is JPEG encoded. The inter-frame coding compresses the differences between frames. Motion compensated prediction technique is utilized. Motion prediction can be applied on image blocks or on video objects. A search for a good match block or object in the reference frame is done. If such block or object is found, their motion vectors are transmitted, as well as the difference of the current frame with the compensated image is also JPEG encoded and sent. There are two types of frames using motion compensated prediction: predicted frame or P-frame, which is coded using only previously decoded frames as reference frames, and Bidirectional predicted frame or B-frame, which is predicted from past and future frames.

In Fig. 1, the left side shows the encoder and the right one shows the decoder. It is noted that the encoder contains in itself a d ecoder to calculate the error term which is the difference of the current frame with the compensated one. Then the decoder takes the reference frame, applies to it the transmitted motion vectors and adds the error term, to finally get the current image.

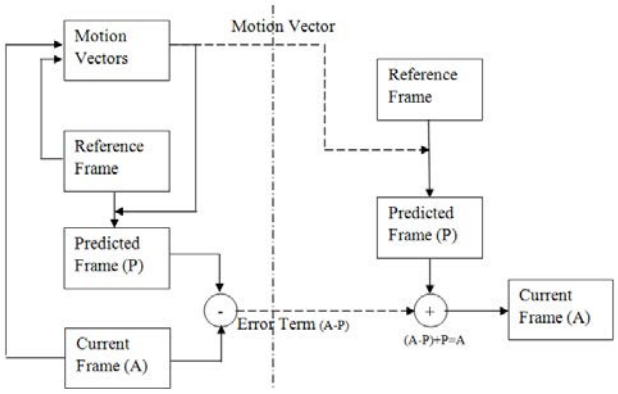

Fig.1. Motion compensation based video compression

Motion estimation is the most computationally expensive operation in the whole compression process. It also influences the performance of the encoder. Hence, in the last decades, researchers take an important interest to this field.

2.1. Motion estimation

There are several methods treating the motion estimation problem. In [18], Laguittonand and Toumoulin presented a motion analysis review. The authors discussed different method dealing with motion estimation as block matching, differential methods, Bayesian filtering, etc. Block matching algorithm is the most basic and the largely used method for video compression. The current frame is divided into a matrix of macro blocks. The latter are compared with corresponding blocks and their adjacent neighbors in the preceding frame. The block that best match the corresponding one in the frame is then chosen and the motion vector is deducted. This movement is calculated for all the macro blocks comprising a frame. An optimal performance for motion estimation is produced using the full search algorithms [19] at the expense of a huge encoding complexity. Other faster algorithms were proposed [19-20] allowing a smaller number of block candidates in the search area such as three step s earch, four step search, diamond search , adaptive rood pattern search, etc. The differential methods are derived in 1981 from the work of Horn and Schunck [21] and Lucas and Kanade [22]. They are based on luminance conservation assumption over time with small movements between two successive images [23-24]. Kalman filtering is a particular case of the Bayesian filtering technique. Given a previous state, kalman filter can produce the current one. This filter is a single step-ahead-prediction method.

In this work, a multi-step-ahead-prediction neural network is used to predict the video object motion in a set of frames. We consider here movies sequences with moving objects

and stationary camera. First of all, we are bringing to localize the video object in a frame and separate it from the background. Thus, we have to estimate the background as well as the video object and trying to stipulate its trajectory throughout the scene.

2.2. Background estimation

Background estimation is usually used for detecting moving objects in a video sequences taken from stationary camera. Moving objects can be detected by the difference of a current frame and a r eference frame, called "background model", or "background image". The background image must be a r epresentation of the sequence with no moving objects. Many different background estimation techniques have been proposed over the last years like a mixture-of-gaussians model [25], background registration technique [26], mean-shift based estimation [27], eigen-backgrounds, temporal median filter, Kernel density estimation, Kernel-based Background Learning [28]. A review on these and other techniques can be found in [29]. A common principle of these techniques is to construct a background model in order to compare it to each new frame for differentiating the regions of unusual motion. Temporal median filter is one of the primary methods to estimate the background model. A median value of the last n frames is considered as the background model. This approach is used in this work in order to estimate the background of video sequence.

2.3. Video object detection, segmentation and tracking

Video compression standards like MPEG-4 use object segmentation principle to identify the moving objects of each frame in a video sequence [30]. Moving object detection from a video sequence is widely used for target tracking purposes. Its objective is to locate the foreground objects in the video sequence, which defines objects of interest. Many approaches have been proposed for video object segmentation. Background subtraction approach is generally used to segment video objects with a stationary camera. The moving object is extracted by computing the difference between the current frame and the estimated background model [31-33]. The difference between two successive frames allows extracting the position and the shape of the moving objects [26]. Optical flow approach detects moving objects in a s cene and generates their correspondence velocity [33]. Other approaches for detecting moving objects are used such as: level sets [34], active contours [35], geodesic active contours [36], visual attention and spatio-temporal information saliency [37]. Combining the spatial segmentation criterion with the temporal one gives more accurate segmentation results. In [38], the author found better results using spatial homogeneity as the primary criteria, which incorporates motion information and luminance simultaneously.

Background subtraction algorithms produce a binary image at each frame in a v ideo sequence representing the foreground mask which corresponds to the moving objects in the video. Several approaches have been proposed to track objects throughout the scene relying on background subtraction. Matching approach is one of the most used to track objects in video sequences [39]. Hence, a correspondence problem is posed to best match objects between successive frames. This problem can be solved by exploiting consistencies in the video objects as position, shape, velocity and appearance. Simple foreground masks features and objects features can be matched between successive frames like areas, bounding boxes and appearance histograms [33]. Also, matching shape models appear in various work as cited in [32, 40]. Connected components in the binary foreground masks are tracked. An object identity is assigned per connected component and is maintained through successive frames. When a new component appears, a new object identity is assigned.

In this work, we will consider the problem of multiple objects tracking in which occlusions, object fragmentation and object merging are not considered. We will consider rigid moving objects with a stationary camera. Tracking problem is still a challenge. Many researches have been proposed to improve this problem solving. In [41], the authors proposed a multiple object tracking using a neural cost function. Real-time motion estimation by object-matching for high-level video representation is proposed by Aishy et al in [42]. A general framework for multi-human tracking using Kalman filter and Fast Mean Shift algorithms is proposed by the authors in [43]. Kalman filters are usually used to make predictions for the subsequent frame and to locate position or to identify associated parameters of the moving video object.

Our video compression scheme is summarized as follows: After estimate the background model of the video sequence with temporal median filter technique, the video objects are extracted using background subtraction algorithm. The segmentation results are enhanced by combining spatial segmentation criterion as 'canny' and 'gradient' filter. Once objects are extracted in each frame, matching approach is applied in order to track these objects throughout the scene. An object identity is assigned for each connected component in the binary foreground masks and is maintained through successive frames. To find the best match object, different consistencies are exploited such as object position, bounding box dimension, and area. The object that closely matches the corresponding one in the reference frame is chosen and then motion vector is deduced. If no match is found, then a

new object identity is assigned. Once motion vector is deducted for each video object in the scene, it will be taken as an input vector to the neural network. The latter has to be trained so as to be adapted to the object motion. Multi-step-ahead-prediction neural network is performed to predict the video object motion in multiple frames. As we have mentioned, in this work we consider movies sequences with moving objects and stationary camera. Thus, at the encoder side, we have to transmit one time the background and the video objects as their initial positions in addition with neural networks parameters. Then, the decoder has to take the transmitted background image, the video objects with their corresponding initial positions. Prediction of motion vector for each video object throughout the scene is performed using multi-step-ahead prediction neural network. Then, each frame is reconstructed by applying on the background image the existing video objects and their associated predicting placements and trajectories over the sequence. In the next section, we detail how neural network can be employed into our video compression approach.

3. Neural network based video compression method

In recent years, neural networks have been successfully applied to video compression. They have been used for frame image compression for example. Kohonen neural network is used to resolve optimization problem of vector quantization method [44]. Feed-forward and locally recurrent neural networks are used in [16] to achieve Quad-tree Segmentation approach. Neural networks have been also used for object segmentation and clustering since Video signals can be viewed as a set of different objects which can be coded independently. To perform video segmentation into different objects, fuzzy neural network is applied to segment the video frames by identifying the background and the foreground. For human image sequences as video conference video, a neuro-fuzzy video segmentation is performed by combining the spatial and the temporal information [45-46]. Moreover, neural networks have been employed for motion estimation. Hierarchical motion estimation is performed by using a Hopfield neural algorithm [47].

In this paper, an object-based video compression using neural networks is proposed. Neural networks are employed to perform prediction of the objects motion throughout the scene. Given an actual position of an object in a frame F_n, the next p positions of this object from frame F_{n+1} to frame F_{n+p} are predicted. Feedforward neural networks are used to predict nonlinear signal since 1987 [48]. Thus, during recent years neural networks have attracted researchers' attention to produce more models for time series prediction. For example, predicting video traffic or video conferencing sequences have been achieved using recurrent neural network [49]. In this work, we propose to apply a recurrent neural network in order to accomplish a multi-step-prediction for object positions in a set of frames.

3.1. Recurrent neural networks

Neural networks can be divided into two categories according to the nature of the connections between different network layers. There are networks with only feedforward connections from the input layer to the output one. These are feedforward neural networks like Multi-layer perceptron (MLP). Networks that present one or more feedback connections, that is they have at least one directed cycle are called recurrent neural networks. This creates an internal state and gives held to interesting dynamic behaviors. Dissimilar to feedforward neural network, recurrent one can employ their internal memory to process arbitrary input sequences. This memory can be exploited for the multi-step prediction by storing previously predicted values to generate new values through time.

Several methods for training a recurrent neural network exist. Backpropagation through time (BPTT) is one the most widely used. BPTT method is based on converting the recurrent network with feedback connections to a feedforward network without any feedback connection by folding the network through Time. The most frequently used training method for feedforward networks is the backpropagation algorithm. BPTT is an adaptation of this training method [50].

3.2. Application recurrent neural network to video compression

In this work, a recurrent neural network is applied to video compression process. Given an initial position of a video object in a certain frame, the role of this neural network is to predict its subsequent positions in the next frames set. A multi-step-ahead prediction is then carried out. At the encoder side, the video sequence is got to be analyzed. Detection and extraction processes of moving objects are performed. Also, tracking scheme is carried out in order to track the movement of each object throughout the scene. Then, a recurrent neural network is trained so as to be adapted to the object motion. This network is composed by three layers (see Fig. 2): the first one is the input layer, the second is the hidden layer and the third one is the output layer.

Input layer constitutes the dynamic memory of the network and serves to memorize the past inputs of the network. This dynamic memory is obtained by the feedback

connection from the output layer to the input one. Information available back in time or past inputs are inserted in a delayed buffer according to the size of the tapped delay line *d* and are discretely shifted as time passes, *Y=y(t),y(t-1),y(t-2),…,y(t-d)*. Past inputs can be external or estimated inputs. For *h*-step-ahead prediction, neural network architecture is used *h* times. For each time step prediction *k (k=1,...,h-1)*, the output $\hat{y}(t+k)$ is memorized in the input layer to estimate the next value (see Fig. 2).

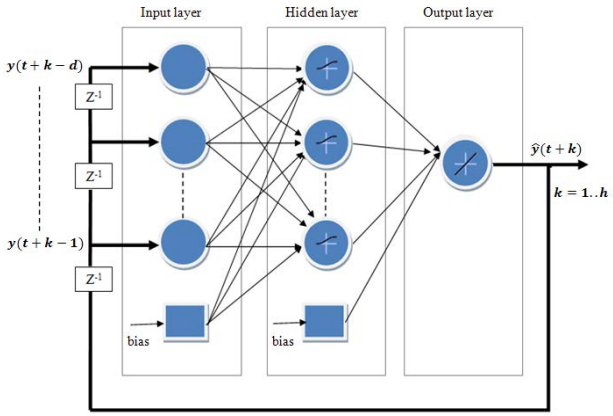

Fig. 2. Proposed neural-network architecture for multi-step-ahead prediction

The hidden layer serves to memorize features of learned examples, increasing then the generalization ability of the network as with feedforward neural network. To train the recurrent neural network, backpropagation through time (BPTT) is employed. BPTT is a gradient-based technique which is used to minimize total error in order to adjust each weight in proportion to its derivative with respect to the error. Two different styles of training exist: incremental training and batch training. In incremental training the weights are updated each time an input, *t*, from the training set is presented to the network. In batch training the weights are only updated after all the inputs are presented. Summed squared error (*SSE*) is used to measure performance function. Each pattern from the training set, *t*, adds to the cost, all output units *k*:

$$E = \frac{1}{2}\sum_{t}^{n}\sum_{k}^{h}[y(t+k) - \hat{y}(t+k)]^2 \quad (1)$$

$$\Delta w = -\eta\frac{\partial E}{\partial w} \quad (2)$$

where *y* is the desired output, $\hat{y}$ is the network output, *n* is the total number of available training samples, *h* is the multistep prediction horizons, i.e. the total number of output nodes and η is the learning rate.

A linear transfer function is employed for the output neurons, while a log-sigmoid transfer function is employed for the neurons in the hidden layer:

$$f(x) = \frac{1}{1+e^{-x}} \quad (3)$$

4. Experimental results and discussion

4.1. Video corpus

The "highway" video sequence is chosen to evaluate the performance of our proposed video compression approach. This sequence is taken from a stationary camera during day time and with same light intensity. The sequence that we consider is composed by 322 frames cropped to 230 x 352 pixels of 24 bits-true color each one. That is, this raw data occupies 625658880 bits, i.e. 74.5844 MO. Fig. 3 depicts the frame number 116 of the original sequence "highway".

Fig. 3 Original image (Frame number 116)

4.2. Video analysis

First of all, we have applied the temporal median filter technique to estimate the background image (see Fig. 4). Then, background subtraction approach is used in order to detect and segment video objects. Background subtraction algorithms produce a binary image at each frame in a video sequence representing the foreground mask which corresponds to the moving objects in the video. We have combined the spatial segmentation with the temporal one in order to improve video-objects segmentation results. Fig. 5 illustrates the detected and segmented video objects in the frame 116.

Fig. 4 Estimated background image

Fig. 5 Detected and segmented video objects in the frame 116

Once objects are extracted in each frame, matching approach is applied in order to track these objects throughout the scene. An object identity is assigned for each connected component in the binary foreground masks and is maintained through successive frames. To find the best match object, different consistencies are exploited such as object position, bounding box dimension, and area. The object that best matches the corresponding one in the reference (previous) frame is chosen and then the motion vector is deduced. If no match is found, a new object identity is assigned. Once motion vector is deducted for each video object in the scene, it will be taken as an input vector to the neural network. The latter has to be trained so as to be adapted to the object motion.

4.3. Recurrent neural network for video compression

We predict objects motion in the highway sequence by using a recurrent neural network. This network is composed by three layers. The first layer has 4 inputs neurons to introduce the values of the centre coordinate as well as the height and the width of the bounding box of the video objects. The hidden layer comports 10 neurons and the output layer comports h sets of 4 output neurons, where h is the multistep prediction horizons. In this work a 10-multi-step-ahead prediction is carried out. The learning rate η is fixed to 0.01 and tapped delay line d is fixed to 6. At the input, all data are normalized to be in the interval [-1 1] and then in the output they are transformed to the real values. The following figures show the first step prediction (Fig. 6), the 5th step prediction (Fig. 7) and the 10th prediction (Fig. 8) of the bounding box width for the second video object detected in the scene.

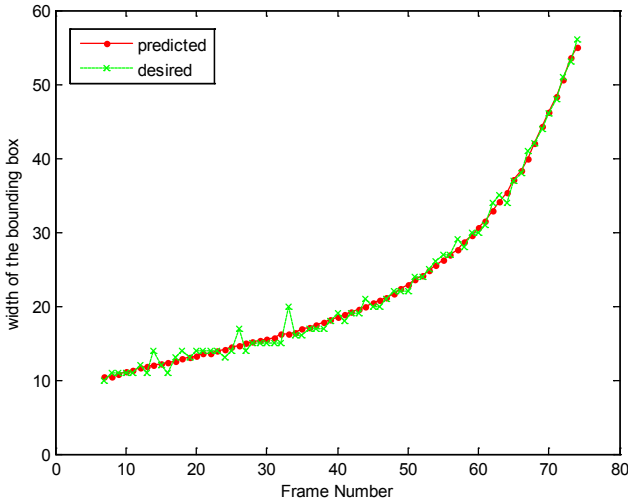

Fig. 6- 1st step prediction of bounding box width for the second video object detected in the scene

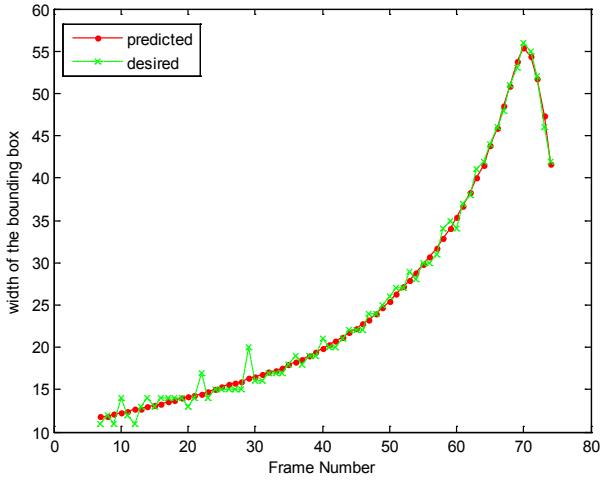

Fig. 7- 5th step prediction of bounding box width for the second video object detected in the scene

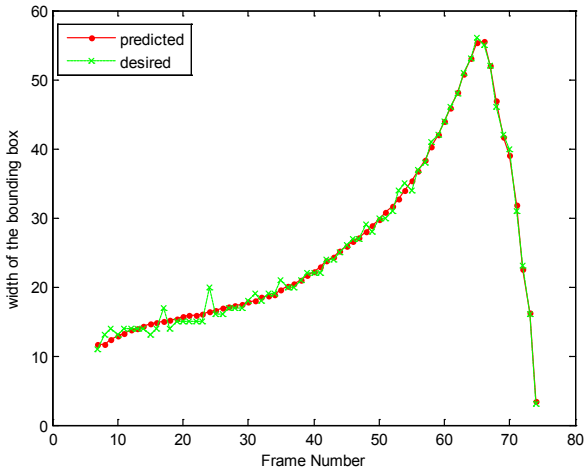

Fig. 8- 10th step prediction of bounding box width for the second video object detected in the scene

At the encoder side, we have to transmit one time the background and the video objects as their initial positions. The video object that has the best resolution is chosen to be transmitted as a pattern and then is resized and placed according to its positions and dimensions over the frames. Furthermore, the decoder has to take the transmitted background image, the video objects with their corresponding initial positions. Prediction of motion vector for each video object throughout the scene is performed using 10-step-ahead prediction neural network. Training the neural network is performed per each video object. Hence, each frame is reconstructed by applying on the background image the existing video objects and their associated predicting placements and dimensions over the sequence. The Fig. 9 illustrates a reconstructed frame by applying the video objects on the background with the respect to their positions and dimensions in a given frame.

Fig. 9- Reconstructed frame number 116

4.4. Performance metrics and evaluation

To analyze the results of our proposed work, Peak Signal to Noise Ratio (*PSNR*) and Compression Ratio are employed. *PSNR* is calculated between the original frame and the reconstructed one:

$$PSNR = 10\ log_{10}\left(\frac{255^2}{mse}\right) \quad (4)$$

Where *mse* is the Mean Square Error defined by:

$$mse = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(y_{i,j} - x_{i,j}\right)^2 \quad (5)$$

where m and n denotes respectively the number of rows and columns in the image and $x_{i,j}$ and $y_{i,j}$ represent the corresponding pixel respectively in the original and reconstructed frame.
Fig. 10 depicts the *PSNR* values of the reconstructed video sequence.

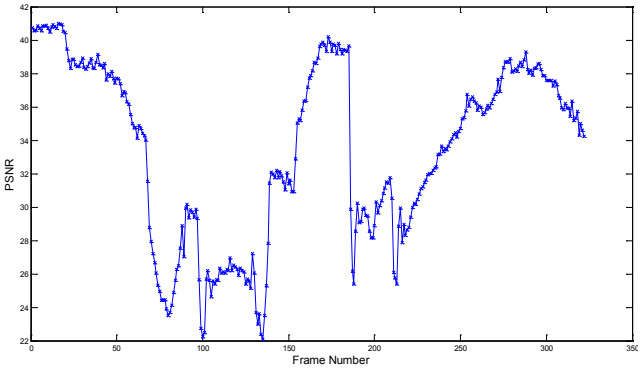

Fig. 10- PSNR values of the reconstructed video sequence

Compression Ratio (CR) is the ratio between the number of bits required to transmit the video sequence before compression and after compression. Before compression the video is coded with 625658880 bits whereas after compression the video is coded only with 2219880 bits, that is mean a CR in the order of 280 (1:280).

4.5. Discussion and analysis

We have obtained a good result in term of compression ratio at the expensive of the quality of the compressed video. As we observe in the Fig. 10, there are well reconstructed frames with high PSNR values and also there exist badly reconstituted frames with law *PSNR* values. This is due to the small illumination changes since we have transmitted one time the background over the video sequence, which has not in reality the same illumination

throughout the scene. Also, a pattern per video object is transmitted to be placed and scaled in the appropriate frame, whereas the video object may change vision according to its distance from the camera. The smallest PSNR values in the curve of Fig. 10 are due since a new object appears in the scene. We can conclude that our proposed approach is a lossy compression method but it is efficient in term of compression ratio. The compression ratio can be highest when the video sequence presents similar motion objects in a long duration scene.

5. Conclusion

We have presented a new object-based video compression approach using a neural network. The latter is used as a tool to predict object motion throughout a video sequence. In order to reduce video data, only the background of the video sequence is transmitted with the different detected video objects as well as their initial properties such as placement and dimensions. Obviously, video sequences are taken here by a stationary camera for better adaption to this approach. Experimental results show the effectiveness of the proposed approach in terms of compression rates despite the deterioration in the quality of the video. However, this approach becomes very efficient for long duration video sequences which present similar moving objects. Furthermore, this approach presents other important features and can be used for other applications such as video summarization and description. As future work, we propose to study the use of 3D object segmentation and reconstruction for video compression purposes in order to enhance results, give better generalization and improve performances.

References

[1] T. Ebrahimi and M. Kunt, "Visual data compression for multimedia applications", Proceedings of the IEEE, vol86, no 6, 1998 , pp. 1109- 1125.
[2] ISO/TEC, "Coding of audio-visual objects. Part 2". ISOAEC 14496-2:Information Technology, 2001.
[3] P. Salembier and F. Marqués, "Region-based representations of image and video: Segmentation tools for multimedia services". IEEE Trans. on Circuits and Systems for Video Technology, vol9, no 8, 1999, pp. 1147-1 169.
[4] ISO/TEC, "Coding of moving pictures and Associated Audio for digital Storage media at up to 1.5 Mbitsls. Part 2", ISO/TEC 11 172-2:1993 Information Technology.
[5] CCITT, SG 15 et COM, 15 R- 16E, "Recommendation H.26 1 Video Codec for audiovisual services at p x 64 kbids", COM 15 R- 16^{E}, 1993.
[6] T. Ouni, W. Ayedi, M. Abid, "New low complexity DCT based video compression method", IEEE Xplore. Restrictions apply, 2009, pp. 202-207.
[7] K. Aizawa, and T. S. Huang, "Model Based Image Coding: Advanced Video Coding techniques for low bit-rate applications". Proc. IEEE, vol 83, no 2, 1995.
[8] D. Saupe, R. Hamzaoui, and H. Hartenstein, "Fractal image compression: An introductory overview", Technical report, Institut fiir Informatik, University of Freiburg, 1996.
[9] O. Egger, P. Fleury, T. Ebrahimi, M. Kunt, "High-Performance Compression of Visual Information-A Tutorial Review-Part I: Still Pictures". Proceedings of the IEEE, vol. 87, no 6. 1999.
[10] L. Torres, and E. Delp, "New Trends in Image and Video Compression", EUSIPCO '2000: 10th European Signal Processing Conference, 5-8 September, Tampere, Finland , 2000.
[11] I. Richardson, "H.264 and MPEG-4 video compression, video coding for next-generation multimedia". USA:Wiley, 2003.
[12] F. Joumana and M. Ralph, "A generalized model for Kalman filtering of motion information in video compression", Int. J.Electron.Commun, 2010, pp. 1046–1054.
[13] C. Kuo, C. Chao, and C. Hsieh, "An efficient motion estimation algorithm for video coding using Kalman filter", Real-Time Imaging, 2002, pp.53–64.
[14] T. Wiegand, J.G. Sullivan, G. Bjntegaard, A. Luthra, "Overview of the H.264/AVC video coding standard", IEEE Trans. Circuits and Syst. Video Technol, vol. 13, July, 2003,pp. 560–576.
[15] L. Zhicheng, Q. Shiyin, and I. Laurent, "Visual attention guided bit allocation in video compression", Image and Vision Computing 29 , 2011, pp. 1–14.
[16] D. Vigliano, R. Parisi, A. Uncini, "Video Compression by Neural Networks", Intelligent Multimedia Processing with Soft Computing, Vol. 168, 2005, pp. 205-234.
[17] D. Salomon, "Data compression", 2004, Springer.
[18] S. Laguitton, and C. Toumoulin, "Analyse de mouvement", Elsevier Masson SAS, 2009, pp. 72-82.
[19] A. Barjatya, "Block Matching Algorithms For Motion Estimation", DIP 6620 Spring, 2004, pp. 1-6.
[20] M. Al-Mualla, C. Canagarajah, and D. Bull Video coding for mobile communications, efficiency, complexity,and resilience. USA: Elsevier Science, 2002.
[21] B. Horn, and B. Schunck, "Determining optical flow", Artif Intell; 17, 1981, pp. 185–203.
[22] BD. Lucas, and T. Kanade, "An iterative image registration technique with an application to stereo vision", Proc DARA Image Underst Workshop, 1981,pp. 21–30.
[23] A. Mitiche, and H. Sekkati, "Optical flow 3D segmentation and interpretation: a variational method with

active curve evolution and level sets", IEEE Trans Pattern Anal Mach Intell;28 , 2006, pp. 18–29.
[24] M. Tagliasacchi, "A genetic algorithm for optical flow estimation", ImageVision Comput;25:14 , 2007, pp. 1–7.
[25] P.W. Power, and J.A. Schoonees, "Understanding Background Mixture Models for Foreground Segmentation", Proceedings Image and Vision Computing New Zealand. 2002.
[26] S.Y. Chien, S.Y. Ma, and L.G. Chen, "Efficient moving object segmentation algorithm using background registration technique", IEEE Transactions on Circuits and Systems for Video Technology, vol. 12, no. 7, 2002, pp. 577–586.
[27] M. Piccardi, and T. Jan, "Mean-shift background image modeling", Image Processing, ICIP '04 International Conference on , vol.5, no. 2004,pp. 3399- 3402 Vol. 5.
[28] H.B. Kashani, S. A. Seyedin, and H. S. Yazdi, "A Novel Approach in Video Scene Background Estimation", International Journal of Computer Theory and Engineering, Vol. 2, No.2, April, 2010, pp. 274-282.
[29] M. Piccardi, "Background subtraction techniques: a review". Proc IEEE Int. Conf. Systems, Man, Cybernetics, 2004, pp. 3099–3104.
[30] T. Sikora, "The MPEG-4 video standard verification model", IEEE Transactions on Circuits Systems for Video Technology, Vol. 7, 1997, pp. 19-31.
[31] J. Sun, W. Zhang, X. Tang, and H. Shum, "Background cut", Proceedings of the European Conference on Computer Vision : s.n. , 2006, pp. 628–641.
[32] D. Koller, J. Weber, and J. Malik, "Robust multiple car tracking with occlusion reasoning", The third European conference on Computer vision, vol. 1,Stockholm, Sweden, 1994, pp. 189–196.
[33] J. Owens, A. Hunter, and E. Fletcher, "A fast model-free morphology-based object tracking algorithm", British Machine Vision Conference, vol. 2, 2002, pp. 767–776.
[34] T. Brox, A. Bruhn, and J. Weickert, "Variational motion segmentation with level sets", European Conferenceon Computer Vision, 2006, pp. 471–483.
[35] M. Yokoyama, and T. Poggio, "Acontour-based moving object detection and tracking", IEEE International Workshop on Visual Surveillance and Performance Evaluation ofTrackingandSurveillance, 2005, pp. 271–276.
[36] W. Fang, and K.L. Chan, "Using statistical shape priors in geodesic active contours for robust object detection", International Conference on Pattern Recognition, 2006, pp. 304–307.
[37] L. Chang, C. Y. Pong, and Q. Guoping, "Object motion detection using information theoretic spatio-temporal saliency", Pattern Recognition 42, 2009, pp. 2897-2906.
[38] P. Salembier, "Morphological Multiscale Segmentation for Image Coding". Signal Processing, vol. 38, 1994, pp. 359-386.
[39] A. Mitiche, and P. Bouthemy, "Computation and analysis of image motion: a synopsis of current problems and methods", Intern. J. Comput. Vis., 19(1), 1996, pp. 29–55.
[40] A. Baumberg, and D. Hogg, "An adaptive eigenshape model", Proceedings of the Sixth British Machine Vision Conference, vol. 1, pp. 87–96.
[41] J. Humphreys and A. Hunter, "Multiple object tracking using a neural cost function", Image and Vision Computing 27, 2009, pp. 417–424.
[42] A. Aishy, M. Amar, and D. Eric, "Real-Time Motion Estimation by Object-Matching for High-Level Video Representation", 2005.
[43] A. Ahmed and T. Kenji, "A General Framework for Multi-Human Tracking using Kalman Filter and Fast Mean Shift Algorithms", Journal of Universal Computer Science, vol. 16, no. 6, 2010, pp. 921-937.
[44] N.M. Nasrabadi, and Y. Feng, "Vector quantization of images based upon Kohonen self organizing feature maps", IEEE Proceeding of international conference of Neural Networks, S.Diego, CA, 1988, pp. 101-108.
[45] S. J. Lee, C. S Ouyang, and S. H. Du, "A neuro-fuzzy approach for segmentation of human objects in image sequences", IEEE Transactions on Systems, Man and Cybernetics, Part B vol33, no3, 2003, pp. 420-437.
[46] G. Acciani, and C. Guaragnella, "Unsupervised NN approach and PCA for background-foreground video segmentation", Proc. ISCAS 2002, Scottsdale, Arizona, USA.
[47] S. S Skrzypkowiak, and V. K. Jain, "Hierarchical video motion estimation using a neural network", Proceedings, Second International Workshop on Digital and Computational Video, 2001, pp. 202-208.
[48] A. Lapedes, and R. Farber, "Nonlinear Signal Processing Using Neural Network: Prediction and System Modeling", Technical Report LA-UR-87-2662, Los Alamos National Lab, 1987.
[49] P.R. Chang, and J.T. Hu, "Optimal nonlinear adaptive prediction and modeling of MPEG video in ATM networks using pipelined recurrent neural networks", IEEE J. Select. Areas Commun., vol. 15. 1997,
[50] P. Werbos, "Backpropagation Trough time : What it does and how to do it", Proceedings IEEE, Vol. 78,1990.

Soumaya Ghorbel received the Engineering degree in Computer Science from the National Engineering School of Sfax- Tunisia in 2005. She received the Master degree in Automatic and Industrial Computer from the same school in 2006. She is currently working on the doctoral degree. Her research interests include pattern classification, character recognition, neural networks, and video compression.

Maher Ben Jemaa obtained his diploma of Engineer in Computer Science from ENSI (Tunisia) in 1989, his Ph.D. from INSA of Rennes (France) in 1993 and his HdR in 2010 from ENIS (Tunisia). He joined the National School of Engineers of Sfax as

Assistant Professor of Computer Science in 1995. He became an Associate Professor in 1997 and he is a professor second degree since March 2011. His current research areas include Pattern Recognition and Fault Tolerance of distributed systems.

Mohamed Chtourou received the Engineering Diploma in electrical engineering from the Ecole Nationale d'Ingénieurs de Sfax-Tunisia in 1989, the Diplôme d'Etudes Aprofondies in Automatic Control from the Institut National des Sciences Appliquées de T oulouse-France in 1990, and the Doctorat in Process Engineering from the Institut National Polytechnique de Toulouse-France in 1993 and the Habilitation Universitaire in Automatic Control from the Ecole Nationale d'Ingénieurs de Sfax-Tunisia in 2002. He is currently a professor in the Department of Electrical Engineering of National School of Engineers of Sfax-University of Sfax-Tunisia. His current research interests include learning algorithms, artificial neural networks and their engineering applications, fuzzy systems, and i ntelligent control. He is author and co-author of more than twenty papers in international journals and of more than 50 papers published in national and international conferences.