Simulation of MPEG Codec for Moving Pictures

Rumen P. Mironov[1] and Lidia A. Nikolova[2]

Abstract – **A method for objective assessment of some neurological diseases (ND) is developed. It is based on the automatic processing and analysis of images of a handwritten script belonging to patients with ND and healthy individuals. In this method connected components are extracted, filtered and analyzed both for the whole script and for each of the automatically separated text-lines. Eight common geometric features describing size, shape and orientation of the handwriting are computed and statistically investigated. The vertical stroke size, aspect ratio and mean slope's angle revealed differences between ND-patients and those from the control group. Also the average stroke width and average stroke number relative to the row length can be used to separate the clinical Parkinsonians from the others. Nevertheless additional features and data are required to obtain reliable evaluation of the handwriting changes due to neurological diseases.**

Keywords – **image processing, video coding, simulation.**

I. Introduction

One of the fastest developing technologies in the field of video and audio compression is the MPEG standard [1]. It was initially oriented towards using in the digital television, but afterwards is approved in the personal computers and is included in the software packages of the operating systems as an optimal tool for interactive video, multimedia and graphics applications. In connection with this a number of sub-standards for video and audio compression were developed as Px64 (H.261, H.263), MJPEG, MPEG-1 to 7 [1-3].

In the presented paper a software model of coder/decoder based on MPEG-2 [4] standard is described, which is designed to work in large number of application areas.

II. Mathematical Description

In MPEG-2 [4] standard there is indexing through levels, which form approximated borders between process power, based on image size and profiles, which limit the algorithm properties.

On Fig. 1 the principle block scheme of MPEG-2 coder is depicted [1], where the different blocks perform the following functions:

A. Discrete Cosine Transform and Inverse Discrete Cosine Transform

The forward and inverse discrete cosine transform of fragments with size of 8×8 is described with the following equations:

$$F(u,v)=\frac{C(u)C(v)}{4}\sum_{i=0}^{7}\sum_{j=0}^{7}f(i,j)\times\cos\left[\frac{(2i+1)u\pi}{16}\right]\cos\left[\frac{(2j+1)v\pi}{16}\right] \quad (1)$$

$$f(i,j)=\frac{1}{4}\sum_{u=0}^{7}\sum_{v=0}^{7}C(u)C(v)F(u,v)\times\cos\left[\frac{(2i+1)u\pi}{16}\right]\cos\left[\frac{(2j+1)v\pi}{16}\right] \quad (2)$$

where: $C(u),C(v)=\begin{cases}1/\sqrt{2} & u,v=0\\ 1 & u,v=1,2,\ldots,7\end{cases}.$

B. Quantisation

The standard permits an equal quantisation step for each DCT coefficient, where the step size can vary for each coefficient and for each macroblock, according to the equation $ss = qf[m,n].qs$. The factor qf depends on the position of the coefficients of the block, and the factor qs is the basic quantisation step. By default there are weighted matrixes $qf[m,n]$ defined in MPEG used for interframe and intraframe blocks.

For intraframe blocks the DC coefficient is proportional to the mean value of the spatial block and it is not quantised with value from the weighted matrix.

C. Variable Length Coding

Experimentally is proven that after quantisation a large number of zeroes are obtained in the blocks of DCT coefficients, especially in the high spatial frequencies. In MPEG this is realized by conversion of the two-dimensional area with size 8x8 in one-dimensional sequence by using zigzag scanning and applying variable length coding on it, which is more effective for long series of zeroes.

In some cases MPEG-2 gives the opportunity for alternative vertical scanning which is often more effective especially applied to the half size frame based DCT.

The basic algorithm for coding uses modified Huffman code. In the standard is permitted using the complement modified Huffman code, which is specially optimized and created for improving the intraframe blocks, which is called alternative intra Huffman code.

D. Motion Estimation

The MPEG standard does not define the method for motion estimation, but methods based on block correspondence

[1]Rumen P. Mironov is with the Faculty of Communication Techniques and Technology, Technical University of Sofia, Kl.Ohridski 8, 1000 Sofia, Bulgaria, E-mail: rpm@vmei.acad.bg

[2]Lidia A. Nikolova is with the Institute of Information Technologies, Bulgarian Academy of Sciences, 1113 Sofia, "Acad. G. Bonchev" Str., Bl.2, E-mail: lnikolova@iit.bas.bg

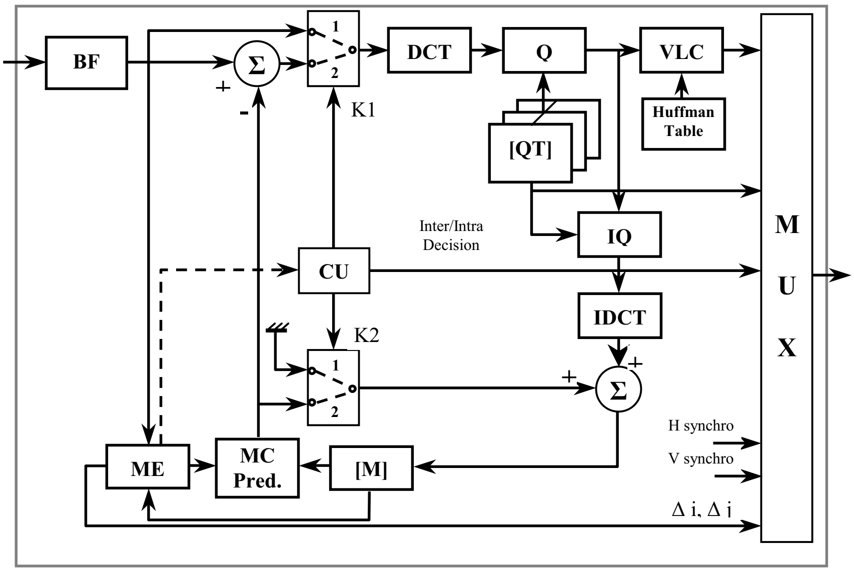


Fig. 1. Block scheme of MPEG-2 coder

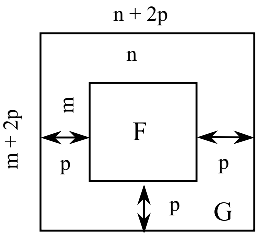


Fig. 2. Motion estimation macroblocks

are preferred. In them block with size $[m \times n]$ is compared to the ones connected with him, in search area with size $[(m + 2p) \times (n + 2p)]$ from previous or following frame. For typical MPEG system the correspondence block (or macroblock) is with size 16×16 pixels ($m = n$=16) and parameter =6, as depicted on Fig. 2, where F is the macroblock in the current frame, while G is the search area in the previous (or following) frame.

In the literature some basic algorithms for estimation of the motion vector are described as: algorithm for full search; three step search algorithm; algorithm with two dimensional logarithm search; algorithm for searching with connected direction; parallel hierarchical one dimensional algorithm for searching and algorithm for modified classification with the pixel differentiation (algorithm with layer structure).

All of them are using different estimation functions [3].

Mean Absolute Difference (MAD), defined as:

$$MAD(dx, dy) =$$
$$= \frac{1}{mn} \sum_{i=-\frac{n}{2}}^{\frac{n}{2}} \sum_{j=-\frac{m}{2}}^{\frac{m}{2}} \left| F(i,j) - G(i+dx, j+dy) \right| , \quad (3)$$

where: $F(i,j)$ is macroblock from the current frame, $G(i,j)$ represent the same macroblock from a previous or a following frame, and (dx, dy) – vector representing the search area, limited in the interval $\{-p, +p\}$.

Mean Square Difference (MSD), which is defined as:

$$MSD(dx, dy) =$$
$$= \frac{1}{mn} \sum_{i=-\frac{n}{2}}^{\frac{n}{2}} \sum_{j=-\frac{m}{2}}^{\frac{m}{2}} \left[F(i,j) - G(i+dx, j+dy) \right]^2 . \quad (4)$$

Cross Correlation Function (CCF), defined as:

$$CCF(dx, dy) =$$
$$= \frac{\sum_i \sum_j F(i,j) G(i+dx, j+dy)}{\left[\sum_i \sum_j F^2(i,j) \right]^{\frac{1}{2}} \left[\sum_i \sum_j G^2(i+dx, j+dy) \right]^{\frac{1}{2}}} . \quad (5)$$

MAD estimation function is appropriate for video applications, because is easy for hardware implementation. The other two estimation functions MSD and CCF might be more effective but are difficult for hardware implementation. For simplifying the calculation complexity of MAD, MSD and CCF estimation functions a simple criterion for correspondence is proposed – pixel differential classification (PDC). PDC criteria is defined as:

$$PDC(dx, dy) = \sum_i \sum_j T(dx, dy, i, j), \quad (6)$$

where:

$$T(dx, dy, i, j) = \begin{cases} 1, & |F(i,j) - G(i+dx, j+dy)| \leqslant t \\ 0, & |F(i,j) - G(i+dx, j+dy)| > t \end{cases}$$

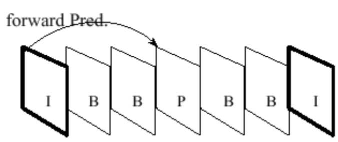


Fig. 3. P-picture coding

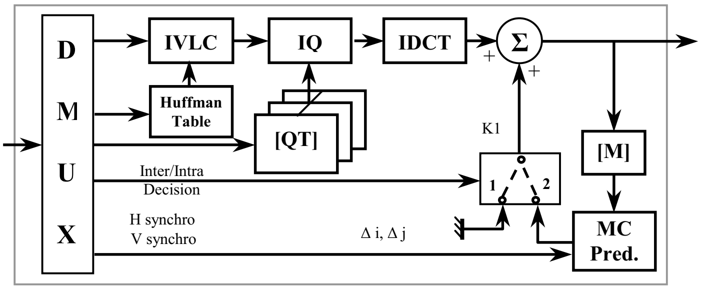


Fig. 4. Block scheme of MPEG-2 decoder

is the binary representation of pixel difference.

In this way each pixel from the macroblock is classified either as a pixel with correspondence (T=1) or as a pixel for which there is no correspondence (T=0). The block, for which PDC function has minimum, is chosen as the block with the maximum correspondence.

E. Intraframe and Interframe coding

In the MPEG standard three types of frames are defined: intra, predicted and bi-directional.

The I-pictures are coded intraframe and represent potential points for random access in compressed video date.

The P-pictures coded using forward prediction, based on the nearest previous I- or P- frames. As the I-, the P-frames are used as predicted reference frames for the next B- and P-frames as is shown on Fig. 3. The difference is that the P-frames using motion compensation reach better compression from the possible one for I frames.

The B-pictures are coded by the bi-directional prediction, using as reference both following and previous frames, as is shown on Fig. 5. The B-pictures deliver the best compression and don't distribute errors, as they are never used as reference frames. The bi-directional prediction leads to minimizing the noise effect by two frames averaging.

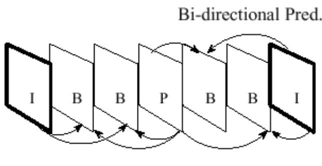


Fig. 5. B-picture coding

On the Fig. 4 [1] the block scheme of the corresponding decoder, which blocks have analogy functions is depicted.

III. Experimental Results

The developed software MPEG-2 codec is realized in correspondence with the basic block schemes, depicted on Fig. 1 and Fig. 4 and support levels LOW and MAIN in the MAIN profile. The different blocks from the schemes are realized as subroutines in the environment MATLAB 5.3 for Windows 9x/2000 platform.

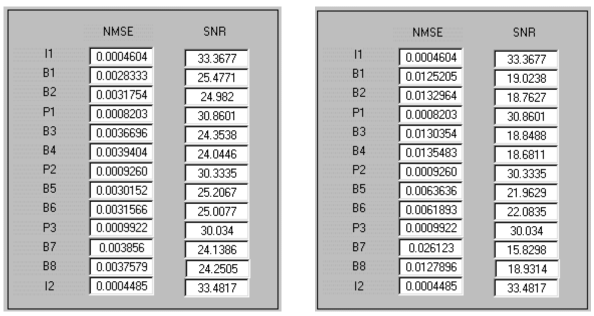
The module which realizes the whole work of the coder and combines all other blocks is ***CODMPG()***. It supports two realizations, which differ by the way of coding of B frames according to: algorithm for coding with bi-directional motion estimation and the algorithm for coding using motion vector interpolation. The other modules are the following:

- $B2 = dctcode(I, mask)$ – forward DCT and quantisation;
- $I2 = dctdecode(B, mask, aman)$ – IDCT and dequantisation;
- $[vectx, vecty] = blkmatch(X1, X2, Mval, Nval, searchlimit)$ – motion estimation, which includes two sub-functions for motion vector estimation using the algorithm for full searching and three step search algorithm;
- $[fcod, Prec] = pframe(frm, Irec, vectx, vecty, Nval, Mval, nx, ny)$ – used for full coding and reconstruction of P frames;
- $[fcod, Irec] = iframe(Itemp)$ – coding and reconstruction of I frames.

The module realizing the whole work and combining all the other blocks is ***DECODMPG()*** and contains the following modules:

- $I2 = dctdecode(B, mask, aman)$ – IDCT and dequantisation;
- $Prec = pframedec(Irec, fcod, vectx, vecty, Nval, Mval, nx, ny)$ – decoding of P-frames;
- $Brec = bframedec(Irec, Prec, Pxy1, Pxy2, Nval, Mval, nx, ny)$ – decoding of B-frames;

The software modeling is performed for video sequences of 13 frames in BMP format, displayed on Fig. 7, where the separate images are ordered in the following order: $\mathbf{I_1}$, $\mathbf{B_1}$,



	NMSE	SNR
I1	0.0004604	33.3677
B1	0.0028333	25.4771
B2	0.0031754	24.982
P1	0.0008203	30.8601
B3	0.0036696	24.3538
B4	0.0039404	24.0446
P2	0.0009260	30.3335
B5	0.0030152	25.2067
B6	0.0031566	25.0077
P3	0.0009922	30.034
B7	0.003856	24.1386
B8	0.0037579	24.2505
I2	0.0004485	33.4817

	NMSE	SNR
I1	0.0004604	33.3677
B1	0.0125205	19.0238
B2	0.0132964	18.7627
P1	0.0008203	30.8601
B3	0.0130354	18.8488
B4	0.0135483	18.6811
P2	0.0009260	30.3335
B5	0.0063636	21.9629
B6	0.0061893	22.0835
P3	0.0009922	30.034
B7	0.026123	15.8298
B8	0.0127896	18.9314
I2	0.0004485	33.4817

Fig. 6. Results for NMSE and SNR for two algorithms

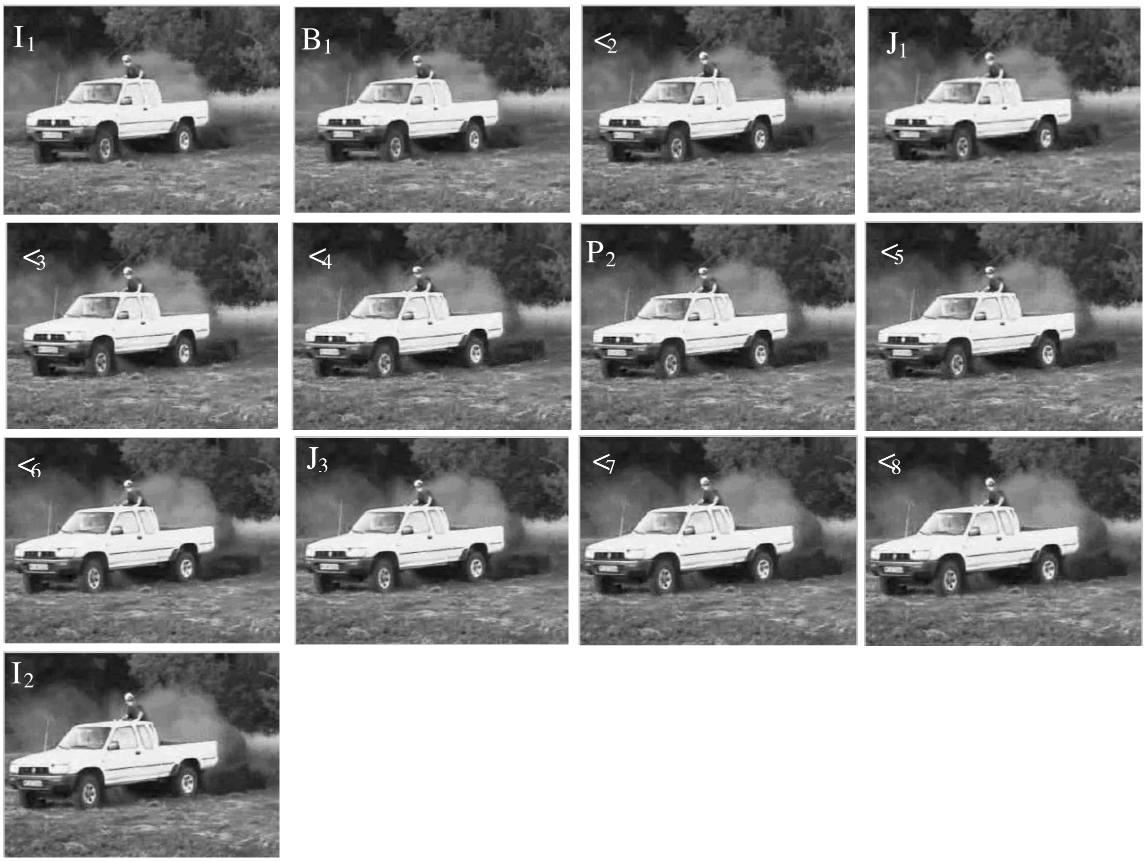


Fig. 7. Original test sequence with 13 frames

$\mathbf{B_2}$, $\mathbf{P_1}$, $\mathbf{B_3}$, $\mathbf{B_4}$, $\mathbf{P_2}$, $\mathbf{B_5}$, $\mathbf{B_6}$, $\mathbf{P_3}$, $\mathbf{B_7}$, $\mathbf{B_8}$, $\mathbf{I_2}$. For testing of the developed software package a PC Pentium MMX 200 Hz / 96 MB RAM with Windows 98 OS is used. The quality estimation is performed subjectively and objectively by estimation of NMSE and SNR for each of the frames. The obtained results are displayed in the output tables, shown on Fig. 6.

The coding time for the first testing sequence is 24 min 47 sec and the decoding time is 8 min 41 sec. The compression coefficient is 40.5175. The coding time for the second sequence is 9 min 8 sec and the decoding time is 8 min 41 sec. The compression coefficient is 41.0985.

IV. Conclusion

Two methods for obtaining the motion vectors of B frames are tested, in order to compare the quality of the video sequence, processed by the coder, calculation complexity and time for coding. From the obtained experimental results is made a conclusion for the advantage of bi-directional motion estimation for B frames for fast changing scenes, by increasing the calculation complexity of the coding process. The method with interpolation is effective for slow changing scenes, for which small distortion in quality is observed, but the number of calculation operations is substantially decreased.

The developed codec is used in laboratory work on the disciplines: "Image and Signal Processing" and "Audio and Video Communication on Internet" and in the experimental work in laboratory "ESVI" in Technical University of Sofia.

References

[1] J.Gibson, T.Berger, T.Lookabaugh. *Digital Compression for Multimedia*. Morgan Kaufmann Publishers, 1998.

[2] J.Ohm. "Encoding and Reconstruction of Multiview Video Objects", IEEE Signal Processing Magazine, May, 1999.

[3] N. Beser, J. Hopkins. Image Compression and Packet video.

[4] T.Sikora. "MPEG Digital Video-Coding Standards", IEEE Signal Processing Magazine, September 1997.

[5] ISO/IEC. "MPEG-2 Description", ISO/IEC JTC1/ SC29/ WG11, July 1996.