Proceedings of the 4th International Conference on Wireless and Optical Communication, WOC03,
July, 2003, Banff, Canada. Editor: Lambertus Hesselink, ISBN: 0-88986-374-1 (38)3 (unformatted version)

EXTREME RATE TRANSCODING FOR DYNAMIC VIDEO RATE ADAPTATION

JAVED I. KHAN & DARSHAN PATEL
Media Communications and Networking Research Laboratory
Department of Math & Computer Science, Kent State University
233 MSB, Kent, OH 44242
javedldpatel@kent.edu

Abstract

This research investigates techniques for extreme scalable video rate transcoding. Traditional requantization based rate transcoding offers a reduction ratio close to 1:5. Unfortunately, the current bandwidth differential between Internet egress points easily exceeds more than 1:60. In this paper we present the analysis of a joint transcoder technique which combines requantization and re-tiling and dramatically extends the transcoding ratio. We demonstrate optimum operation point based on the joint error and entropy analysis of the system, and share some experiments based on live video transcoding performance.

Keywords: video, transcoding, adaptation.

1 Introduction

In recent years the asymmetry of Internet has grown enormously. It is well known that the Internet backbone speed is doubling at every 9-12 months. However, with it is also increasing the Internet asymmetry. The LAN speed has increased quite slower doubling approximately every 3-4 years. With the advent of wireless integrated low power devices, interestingly the low range is also now expected to take a dip. Even using today's egress capacity asymmetry, seamless video transmission will require at least 30-60 times rate adaptation ability, much above what is achievable today. The scalability profiles [1] designed within the current video coding standard indicates that this was not anticipated. The initial expectation seems to that two or three level static scalability would be enough. Unfortunately, the current state of the Internet asymmetry indicates that a much dynamic and a much wider range of video scalability will be required. Though, a wide volume of analytical work exists on techniques for first step data compression, but relatively little work has been reported on optimum transcoding. Interestingly, despite some commonality, quite a few issues in transcoding are different from that in encoding. Encoding is a single step operation, where the end target rate is known. However, dynamic scalable video communication contemplated for very large networks (such as the Internet) requires that a video have to be re-coded a second time to match the requestor's line rate. Lossy compression is a nonlinear process thus piece-wise optimization does not add well. Even if a single step is optimized for the best quality for a given rate (or the best compression for a given quality), when the video undergoes a second stage compression then single stage optimality analysis does not hold.

Compared to video encoding, dynamic **video rate transcoding** is a relatively new area. One of the earliest works is due to Nishitani [2] which discussed ADPCM/PCM transcoding for lossless coding. Several other works investigated faster architectures for requantization based techniques [3,4,5]. Several researches investigated closely associated problems such as buffer and delay reduction in transcoding [6]. Speed was still the major concern. In 1998 Assuncao and Ghanbari [12] presented a DCT domain based requantization technique with highly reduced transcoding complexity. Several works [5,7] also suggested optimization/elimination of motion vector computation in the transcoding stage for complexity reduction. These partially coding schemes provided substantial increase in speed. However, a problem with these partially decoding transcoder is the drift-problem. Drift problem arises from gradual accumulation of errors from a sequence of predictive frames (particularly problematic for large GOP). Youn, and Sun [8] analyzed the drift problem of the proposed fast architectures, and strongly suggested re-investigation of pixel-domain transcoding in favor of their drift-free performance and flexibility in transcoding. Bjork and Chrisopoulos [9] investigated the transcoding in the light of MPEG-7, and evaluated an object based transcoding resolution reduction mode besides requatization based transcoding.

A Rate transcoder reduces an already compressed bit stream into a new bitstream with lower rate. We will refer to this rate as the *transcoding ratio*. The initial stream is expected to be already well compressed. Also the output stream is expected to maintain reasonably perceivable quality.

Notably, in the previous techniques, except [8], the principal means of such rate control is *requantization*. However, our investigation shows that the maximum transcoding ratio offered by requatization is in the range of 1:5. It is fundamentally limited if quantization is the only means used Few of the cited techniques considered

only 20% reduction. Even the resolution reduction technique in [8] studied only the limited scalability range of 120~136 kbps/s to 61~68 kbps stream by fixed 1:2 transcoding ratio. Whereas seamless video transmission will require at least 30-60 times rate adaptation ability even if we consider the current level of egress asymmetry in the Internet—far exceeding what is achievable by requantization alone. In contrast, in this research we investigate how the transcoding ratio can be extended for order of magnitude larger scalability suitable for the Internet.

In this research we analyze a new transcoding architecture which can potentially extend the range by another factor of ten. This new architecture combines the technique of sample tiling, which drastically reduces block header overhead. However, we observe that in lower transcoding ratio quantization based reduction seems to demonstrate superior performance. However, we particularly note that at the higher end the re-quantization looses becomes excessively. Indeed, the tiling approach becomes more important for wider scalability. Indeed, if compared one to one, near the 1/10 transcoding ratio—the extreme end of capability of quantization based transcoders, the performance of tiling based requantization becomes higher than quantization alone rate reduction. The approach proposed here investigates the use of free scale based tiling and joint quantization for achieving higher performance than what is achievable by any individual scheme and at the same time extends the scalability range by order of magnitude. We provide analysis of the proposed transcoder and also demonstrate optimum choice of requantization and the degree of sample tiling for target outgoing line rates.

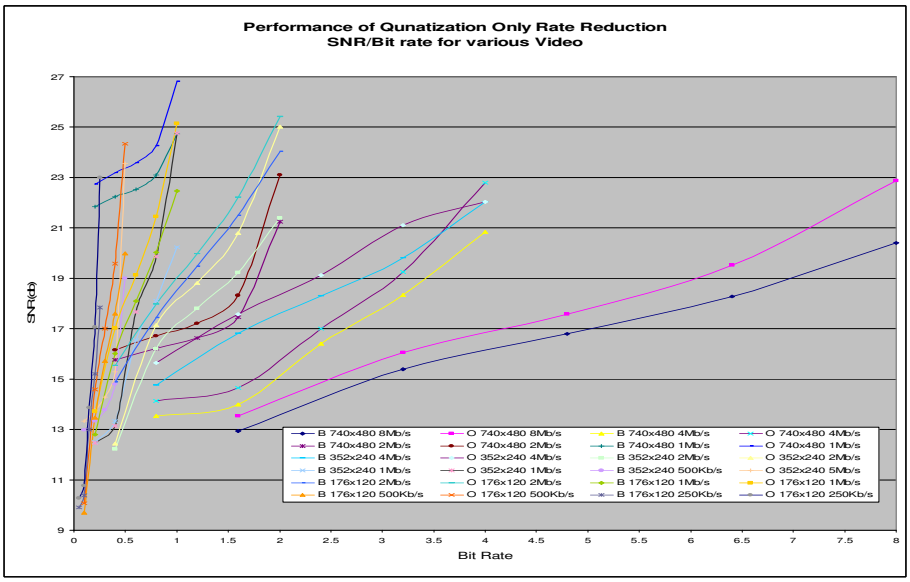

Fig- 1 General Transcoding Ratio

1.1 MPEG-2 scalability overhead:

Each of the 64 MPEG-2 DCT coefficients is encoded with 3-24 bits VLC coding. Each block contains about 2-24 bits of flag overhead (dct_dc_size_luminance, dc_dc_differential etc.). In addition each Macro block adds about ~300 bits of other structures (such as macroblock modes, quantizer_scale_code, motion vectors, coded block pattern, etc.) contributing to the overhead. In 4:2:0 format each MB contains 6 blocks. Even ignoring the overheads due to slice and picture headers the above accounts for somewhere from 4.9% to 40% overhead bits limiting the scope of re-quatization based scalability. The impact of overhead on scalability is shown in Fig-1 experiment. Here several videos with various initial encoding rates were fed into a TM-5 quantization only transcoder with various target bit rate. The graphs plot the corresponding SNRs. As can be seen than the reduction eventually suspended. Despite the specified target rate, the video could not be compressed further because of the overhead.

1.2 Quantization Model:

In transcoding the operation that takes place is requantization is slightly different from basic one step quantization. First we define the basic process. The quatization process in video transmission involves two complementary operations *quantization* (Q) and *dequantization* (DQ). Quantization refers to a process of mapping an incoming real-values sample on a discrete number $n \in [-N, +N]$. It partitions the real valued numbers into 2N+1 disjunct quatization intervals θ_i. An incoming sample x is mapped on the discrete number n if it falls in the interval θ_n. The intervals θ_n are represented by their decision thresholds t_n. For n>0 we will denote the intervals by $\theta_n = [t_{n-1}, t_n]$. For n=0 the interval $\theta_n = [t_{n-1}, t_n]$. For n<0, the intervals $\theta_n=[-t_n, -t_{n-1}]$. We assume that the quantizer is symmetric around zero. The dequantization DQ refers to the inverse lossy process of mapping an incoming discrete number $n \in [-N, +N]$ on a real valued number $\hat{x}$. Each incoming discrete number is mapped on a real-valued number $\hat{x} = r_n$. These real-valued numbers are referred to as the *representation levels* of the de-quantizers. For optimum performance both the quanatizer and the dequantizer have to be designed jointly. Though, in the general case, the representation levels do not need to be equidistant, however, practical quantizers (such as H.261/263, MPEG-1/2) are uniform. Let Δ denote the step size, then the representation levels are:

$$r_n = n \cdot \nabla \qquad ..(1)$$

Most transport protocols such as MPEG-2 strictly define the dequantizer, and thus leaves only the quantizer decision thresholds to be designed by individual encoding

algorithms. For example, the quantizer used inside the MPEG-2 TM5 [1] encoder places the thresholds 5/8 between two representation levels that is for n≥0:

$$t_n = \nabla \cdot \left(n + \frac{5}{8} \right) \quad ..(2)$$

This is an entropy constrained memory less quantizer with regular spacing of the decision thresholds, which makes it easier to implement and its performance is very close to that of 'optimum quantizer'.

1.3 Transcoder Quantization Model:

Existing video (MPEG-2) quantization models have been designed and optimized with single step encoding in mind. However, dynamic scalability implies *cascade quantization*. The performance of cascade quantization is not the same like that of single step. Here the decision levels of the top step effective quantizer is determined by the representation levels of the first quantizer those falls within the decision levels of the second quantizer It can be shown that the effective decisions levels of a cascaded quantizer are given by:

$$\theta_m^{(e)} = \bigcup_{r_n^{(1)} \in \theta_m^{(2)}} \theta_n^{(1)} \quad ..(3)$$

If the step sizes of the initial and second stage transcoders are respectively Δ_1 and Δ_2, (typically the later is larger) then the decision intervals of the effective quantizer no longer remain uniform. These are then given by:

$$t_m^{(e)} = \nabla_1 \left(\left\lfloor \frac{\nabla_2}{\nabla_1} \left(m + \frac{5}{8} \right) \right\rfloor + \frac{5}{8} \right) \quad ..(4)$$

This in different from the result if the quantization with decision interval Δ_2 were done directly to the second quantizer. The representation levels of the effective quantizer however, are given by the decision levels of the second quantizer. But, the effective step size of the joint quantizer is no longer equal. The effective step size is bounded by:

$$\max\left\{\theta_m^{(e)}\right\} = \max(t_m^{(e)} - t_{m-1}^{(e)}) \quad ..(5)$$
$$\Delta_2 - \Delta_1 \le \theta_m^{(e)} \le \Delta_2 + \Delta_1$$

Also, the representation levels are not placed at uniform distance between effective decision boundaries.

1.4 Transcoder Tiling Model:

Similar to quantization, the tiling process consists of two operations, the *tiling* (T) and *inverse tiling* (DT). In tiling generally k spatially adjacent pixels from the base frame are merged into one sample called *tile*. The *tile size* is denoted by k. The tiling operation can be represented as following where the base frame F of size hxw is converted into a tiled frame F_T $h_t x w_t$ of smaller dimension using tile filter W_k:

$$\overline{F}_T = W_k \cdot F \text{ and } \mathrm{k} = \frac{\mathrm{h.w}}{\mathrm{h_t.w_t}} \quad ..(6)$$

The inverse tiling operation performs an upsampling of the tiles and returns the frame to original size. It can be defined by the following where the inverse tiling filter computes an estimate of the frame.

$$\tilde{F} = W^{-1}{}_k \cdot \overline{F}_T \quad ..(7)$$

Several techniques for tiling have been investigated under image scaling techniques. Such as *digital differential analyzer* (DDA). For our case we use a *linear surface approximation* (LSAT) based 2D DDA tiling process based on the DDA suggested by [13]. It is a weighted average based on the surface area overlap. Each tile or pixel has rectangular area coverage. If a_i is the area covered by a pixel x_i and a_T is the area covered by the tile T, then the value of the tile T is determined by:

$$\overline{x}_T = \frac{\sum_{i \in F} (a_i \cap a_T) \cdot x_i}{\sum_{i \in F} (a_i \cap a_T)} \quad ..(8)$$

The corresponding inverse tiling operation is a 2-dimensional linear interpolation (or its computationally optimized version). A fine pixel x_i is recomputed from the four tiles closest to it in its four corners denoted as it's *corner set* $C(i) = \{\overline{x}_{i,UL}, \overline{x}_{i,UR}, \overline{x}_{i,BL}, \overline{x}_{i,BR}\}$. The interpolation is weighted based on the inverse of distances between the centers of the tiles and pixels. Each of the pixels inside the four tile centers divides the space into four inside rectangles. The interpolated pixel values are computed from the areas of these four rectangles. Let $\overline{x}_t$ is a tile in the corner set C(i) of a pixel. Let the area covered by the rectangle defined by C(i) is A_T and $a_{i,t}$ is the area of the inside rectangle closest to $\overline{x}_t$ (with corners $[\overline{x}_{i,t}, x_i]$) and $a_{i,\bar{t}}$ is the area of the inside rectangle furthest, then the interpolation of x_i is given by:

$$\tilde{x}_i = \frac{\sum_{t}^{C(i)} \overline{x}_t . (A_T \cdot a_{i,\bar{t}})}{4 A_T} \quad ..(9)$$

LSAT tiling process enables reduction of video frame size beyond what can be achieved by quantization. The LSAT mechanism is known to create zero error for linearly distributed pixels. However, for curve surfaces it creates a gradual smoothing effect and just like the traditional quantization.

1.5 Joint Error Analysis

Now we will provide a joint analysis of the distortion created by the combined use of the tiling and quantization processes that can be used by the rate transcoder. When $f_{init}(x)$ is the distribution of the original samples, the error created by the first stage quantization is given by:

$$\sigma_{Q1}^2 = \int_{\theta n} (x - r_n)^2 \cdot f_{init}(x) \cdot dx \quad ..(10)$$

The error created by tiling is given by the variance of the distortion created by the process. Here $f_{tile}(x)$ is the probability distribution function of the tiled values, and can be determined from the TM5 operations for certain distributions. In LSAT interpolation the loss occurs during the initial linear approximation of the surface defining the pixel values in the tile area. Thus the error created by tiling can be considered to be bounded by the variance of the distortion created by the piece-wise linear approximation in the tiling operation and is given by the following:

$$\sigma_T^2 = \int_T (x_i - \hat{x}_{T\ni i})^2 \cdot f_{tile}(x) \cdot dx \quad ..(11)$$

The error added by the second stage quantization of the tiles is given by:

$$\sigma_{Q2}^2 = \sum_{all\ \theta_m} \int_{\theta_m} (\bar{x}_t - r_m)^2 \cdot f_{tile}(\bar{x}_t) \cdot d\bar{x} \quad ..(12)$$

The distortion can be computed in a logarithmic format by means of PSNR. The *peak-signal-to-noise-ratio* (PSNR) is defined as:

$$PSNR = 10 \log_{10}\left(\frac{255^2}{\sigma^2}\right) \quad ..(13)$$

The value 255 refers to the maximum possible value of the pixel value.

1.6 Joint Entropy Analysis

The bitrate resulting from the re-quantization and tiling process can be estimated based on entropy analysis. For requantization the probability that x is in the m-th tiling interval is given by:

$$p_m = \int_{\theta_m} f(\bar{x}) \cdot d\bar{x} \quad ..(14)$$

These intervals are uneven however with bound 1. The entropy of each symbol is then given by:

$$H = -\sum_m p_m \cdot \log_2 p_m \quad \text{bits/symbol} \quad ..(15)$$

Let us assume that F is the initial samples/frame, k is the tiling factor (samples/tile), B is the number of tiles in each encoding block (Macro Block for MPEG) and H is the amortized coding block overhead. Then the bit requirement for each frame due to quantization is given by:

$$\Pi = \frac{B \cdot H}{k} - \frac{\alpha \cdot F}{k} \sum_m p_m \cdot \log_2 p \quad \text{bits/frame} \quad ..(16)$$

Here α (<1) is the compression factor due to non quantization techniques used (such as variable length decoding).

1.7 Mute- Quantization-only Reduction

If we use only quantization as a means of reduction, then this Q-only schemes quantization factor for any given outgoing data rate R can be estimated from equation (20) with k=1.

$$R = \frac{B \cdot H}{1} - \frac{\alpha \cdot F}{1} \log_2 \frac{1}{m} \quad \text{bits/frame} \quad ..(17)$$

Thus:

$$m_{Q-only} = 2^{\frac{R - BH}{\alpha \cdot F}} \quad ..(18)$$

1.8 Aggressive- Tiling-only Reduction

In opposite, extreme a tiling only (T-only) reduction can reduce samples without directly reducing the quantization factor. Since, tiling proportionately reduces the header as well as samples, the tiling factor can be computed simply as a ratio of incoming and outgoing bit rate.

$$k_{opt} = \frac{R_{init}}{R} \quad ..(19)$$

1.9 Optimum Hybrid Reduction

However, the least distortion best results can be obtained if we use a Hybrid scheme which prudently uses both tiling as well as quantization. Now we demonstrate the optimum choice of the tiling factor and the number of quantization levels. The optimality is defined in the sense where for a given bit rate the overall distortion can be minimized. We show the solution for *uniform* sample distribution and *triangulated linear sample surface* approximation. We will track the bounds on the maximum distortion that can be generated in each of the processes.

Let s is the average *slope-factor* of the sample surface. The average distortion generated by the tiling process is given by:

$$E = |E_{Q1}| + |E_S| + |E_{Q2}| \quad ..(20)$$

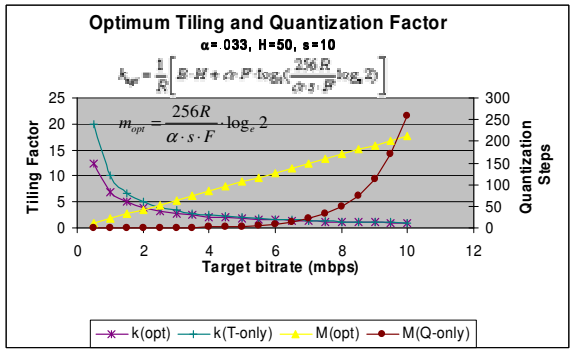

Fig- 2 Optimum Tiling Factor

Let R is the choice of bit-allocation per frame. The number of first stage quatization levels (n) is not selectable at transcoding stage. We can solve for minimum E by:

$$\left.\frac{dE}{dm}\right|_{m=m_{opt}} = 0$$

This gives the optimum quantization steps:

$$m_{opt} = \frac{256R}{\alpha \cdot s \cdot F} \cdot \log_e 2 \qquad ..(21)$$

And the optimum tiling factor:

$$k_{opt} = \frac{1}{R}\left[B \cdot H + \alpha \cdot F \cdot \log_2\left(\frac{256R}{\alpha \cdot s \cdot F} \log_e 2\right)\right] \qquad ..(22)$$

In Fig-2 we show the predicted optimum tiling factor (k_{opt}) and quantization steps for various target reduction rates (m_{opt}). Let *d* is the sample density. For 4:4:4, 4:2:2, and 4:2:0 video *d* is respectively 3, 2 and 1.5 samples/pixel. Thus for a frame size of X x Y pixels, F= X.Y/d samples/frame. We started with an initial 256 step quantizes 740x480 sized 10 Mbps encoded high quality 4:2:0 MPEG-2 video. The plot shows both the predicted optimum *m* and *k* for various reductions (encoding α=0.033, H=50 bits, and s=10).

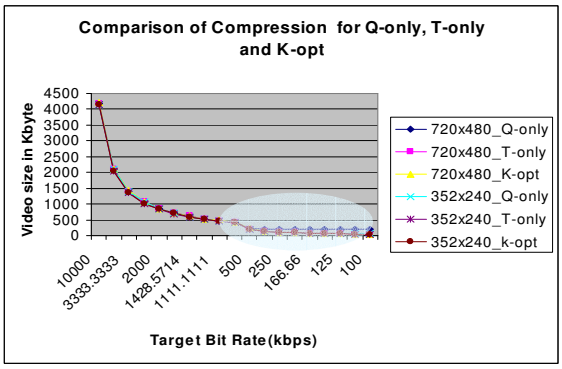

Fig- 3(a) Reduction Efficacy

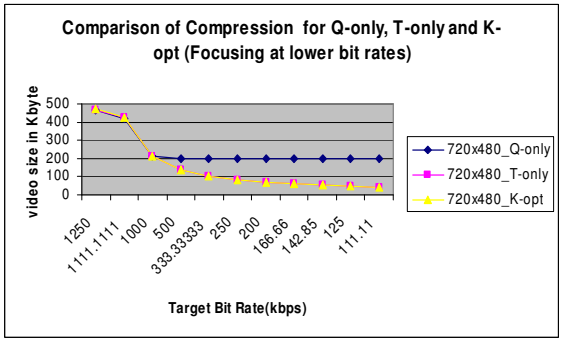

Fig- 3(b) Reduction Efficacy (small range)

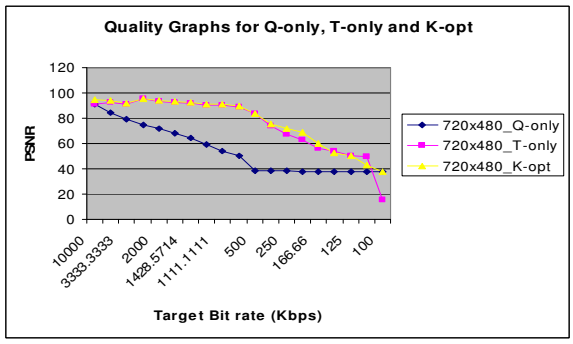

Fig- 4 Quality Graph

2 Performance Analysis

We have developed the test system so that it can operate in all four modes [10]. Q-only, T-only, controlled-Hybrid, auto-optimized Hybrid: In Q-only mode the tiler process is switched off. All compression is left to the quatization. In T-only mode, the tiler is switched on and all compression is done by tiling $k=R_{out}/R_{in}$. In controlled-Hybrid mode the user specifies the tiling factor and the remaining encoding is done by the CBR quantization. In optimized Hybrid mode the optimization algorithm is used to divide the compression between tiling and quantization. We use the tiling factor selected by this formula and let the TM-5 quantization handle the rest. Fig-3(a) shows the rate reduction performance with specified target bit rate over a broader range. As can be seen the reduction works well at the initial stage.

How did these methods performed at extreme range? To see the effect of joint optimization in Fig-3(b) we plotted the same data in a narrower range. As can be seen that indeed the Q-only scheme lost scalability after a while, while the K-optimum matched the performance of the T-only scheme. Fig-4 shows the corresponding SNR quality of this test video. Clearly, the K-opt scheme performed much better than the Q-only scheme. Compared to T-only at the similar bit-rate and quality it produced much larger picture.

Fig-5 finally shows the visual impact when all three pictures were scaled to the same size. This shows one sample frame from a video which was compressed about 10 times. The top figure is the original. The loss of detail in the T-only scheme is visible in the human hand area.

3 Conclusions & Current Work

Traditional video transcoding schemes use only re-quantization schemes for rate reduction. However, due to the nature of the coding scheme a requantization based scalability scheme faces fundamental limits. In this paper we have presented a hybrid scheme which combines tiling with re-quantization. The proposed hybrid technique can extend the scalability range by about a factor of 10. However, it is natural that even tiling based technique will face limits. We are currently investigating next generation

techniques that will further extend the scalability range [11] using content analytic technique.

The work is currently being funded by DARPA Research Grant F30602-99-1-0515 under its Active Network initiative.

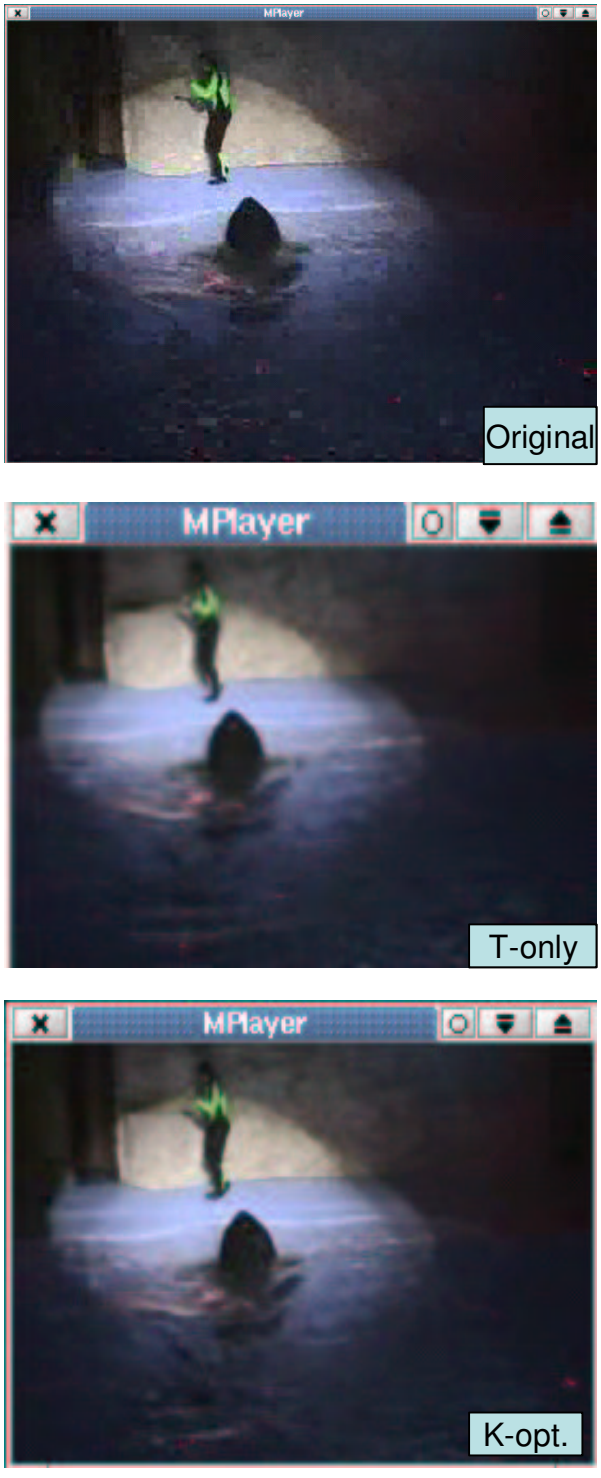

Fig-5 all three figures scaled back to same size after compression

.

4 References:

[1] Information Technology- Generic Coding of Moving Pictures and Associated Audio Information: Video, ISO/IEC International Standard 13818-2, June 1996.

[2] T. Nishitani, Tandem Tanscoding without Distortion Accumulation, IEEE trans. On Comm, Vol COMM-34, no 3, March 1986, pp.278-284.

[3] Morrison D. G., M. E. Nilson and M. Ghanbari, Reduction of the Bit-rate of Compressed ideo while in its Coded Firm, 6th Intl. Workshop on Packet Video, paper D.17.1, Portland, Orego, September 1994.

[4] W. K. Sun and J. W. Zdepski, Architecture for MPEG compressed bitstream scaling, IEEE Transactions on Circuit Systems Video tech, 6(2), August 1996, pp.191-199.

[5] G. Keesman, R. Hellinghuizen, F. Hoeksema, & G. Heideman, "Transcoding of MPEG Bitstreams," Signal Processing Image Comm., vol. 8, pp. 481-500, 1996.

[6] Kan, Kou-Sou; Fan, Kuo-Chin, Video transcoding architecture with minimum buffer requirement for compressed MPEG-2 bitstream Signal Processing, Volume: 67, Issue: 2, , pp. 223-235, June 18, 1998

[7] J. Youn, M.T. Sun, and C.W. Lin, "Motion Vector Refinement for High Performance Transcoding," IEEE, Transactions on Multimedia, Vol. 1, No. 1, pp.30-40, March 1999.

[8] J. Youn, M.T. Sun, Video Transcoding with H.263 Bit-Streams, Journal of Visual Communication and Image Representation Volume: 11, Issue: 4, December 2000, pp. 385 – 403

[9] Niklas Björk and Charilaos Christopoulos, Video transcoding for universal multimedia acces; Proceedings on ACM multimedia 2000 workshops, 2000, Pages 75 - 79

[10] Khan, Javed I., Darhsan Patel, Wansik Oh, Seung-su Yang, Oleg Komogortsev, and Qiong Gu, Architectural Overview of Motion Vector Reuse Mechanism in MPEG-2 Transcoding, Technical Report TR2001-01-01, Kent State University, [available at URL http://medianet.kent. edu/ technicalreports.html, also mirrored at http:// bristi.facnet.mcs.kent.edu/medianet] January, 2001]

[11] Javed I. Khan and Zhong Guo, Flock-of-Bird Algorithm for Fast Motion Based Object Tracking and Transcoding in Video Streaming, Proceedings of the 13th IEEE International Packet Video Workshop 2003, Nantes, France, April 2003 [URL: http://www.medianet.kent.edu/publications/PV2003DL-vodobject-KZ.pdf]

[12] P. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit rate reduction of MPEG-2 bit streams," Trans. On Circuits Syst. Video Technol., vol. 8, no. 8, pp. 953-967, 1998.

[13] Dean Clark, A 2-D DDA Algorithm for Fast Image Scaling, Dr. Dobb's Jounral, April 1997.