Scaling the Evolutionary Models for Signal Processing System Optimization with Applications in Digital Video Processing

Walid S. Ibrahim Ali
Philips Research, Briarcliff Manor, New York

Abstract

Video processing algorithms tend to improve over time in terms of image quality while increasing in implementation complexity. Generally, video algorithms are developed and evaluated in isolation from the video processing system of which they will be a part, in a consumer product. The final image quality obtained by that system, however, strongly depends on the interaction of its constituent algorithms. Current methods for optimizing the overall image quality are ad-hoc, time consuming and don't guarantee the best possible result. In this paper we propose a scalable method for optimizing a video, taking into consideration the possibility of adding/removing different components to this video system. Our method utilizes *genetic algorithms* (GAs), which evolves toward the optimum system configuration (the best image quality). GAs are heuristic optimization search methods, which when driven by an objective video quality metric, finds the optimum system configuration.

1. Introduction

Improving image quality is the backbone of a highly competitive display industry. As video systems are constantly evolving; new video/image processing algorithms are constantly introduced and older ones are being refined. However, individual algorithms are generally developed in isolation, but eventually get implemented as part of a larger system, e.g., a television set, in which they interact with other algorithms. Consequently, the final image quality obtained by a chain of video processing modules in a television system depends on the interaction of its constituent algorithms. This interaction depends on the parameter tuning for each module, the amount of data (bus width, bit precision) being transferred between cascaded modules as well as the order of the cascaded modules in the video processing chain. Ad hoc techniques have been adopted for a long time to come up with the best settings for these systems. However, a thorough analysis and a formal simulation environment of this interaction is required, in order to find the optimal module order, the best tuning of the modules' parameters and the bit precision among the video processing modules.

In this paper, we introduce an automated procedure to tune a set of video processing modules in order to obtain the best perceptual video quality. Parameter tuning, modules' order as well as the data precision among different modules will be addressed. We examine a case study (a simplified video processing chain) consisting of a noise reduction and a sharpness enhancement module. The effect of altering their order, their parameter settings as well as the bit precision between them is automatically analyzed and illustrated. Automatic means of measuring the resultant video quality and a structured procedure to maximize the correlation between these objective measures and human perception is introduced.

This paper is organized as follows: section two describes the cost function, which is the automatic means of video quality assessment. Section three describes the evolution process setup and the gene's construction for a random video processing system. Section four shows the structure and exact experimental procedure for a simplified video processing system. The results of running a video system design using the GAs and the objective image quality metric are introduced in section five. In section six, we make our concluding remarks and propose directions for future developments and research.

2. Cost Function: A Composite Objective Image Quality METRIC

Evaluation of video quality has always been achieved using subjective methods.[9] Since the subjective results vary according to the variability between the viewing audiences, subjective results, which are solely based on perception, have to be statistically post-processed in order to remove the ambiguity resulting from the non-deterministic nature of these results. Linear and Non-linear heuristic statistical models[14] have been proposed to normalize these results, and come up with certain figures of merit to represent the goodness/degradation of the video quality. However, relying on human evaluation is expensive and sometimes impossible to adopt. Thus, a need emerges for automatic methods to evaluate video quality. Automatic (objective) means of assessing the video quality are evaluated by the highest degree of correlation they achieve with subjective testing.[10] The higher the correlation, the better the objective method is.

Different methods are investigated for objective image quality measurement.[4] They vary widely in complexity and performance. They can be categorized in many different ways: measuring traditional analog vs. digital artifacts, measuring the general perceptual quality of a video sequence vs. measuring a specific artifact only, and finally still image (frame/field) evaluation vs. temporal evaluation. Nine models were proposed to the Video Quality Expert Group (VQEG).[14] They varied in performance and complexity.[17,1,4,2] Some methods performed well under certain conditions but failed under others, e.g., peak signal to noise ratio (PSNR) is a good method to measure white noise presence but is not very suitable for measuring coding errors like blocking. We propose a composite scalable objective metric, which consists of a set of metrics, each of which is geared toward measuring a certain feature of the video sequence. Each of these *n* metrics gives a reading, f_i, $(1 \leq i \leq n)$, which measures a certain feature of the video sequence. These readings are weighted by a weight factor each, w_i $(1 \leq i \leq n)$ and linearly combined in order to maximize the correlation factor (R) between human perceptual models and the composite objective measure *F*.

$$F = \max_R \left\{ \sum_{i=1}^{n} w_i f_i \right\} \qquad (1)$$

Figure 1 gives a schematic diagram of the system we adopted for automatically evaluating video sequence image quality.

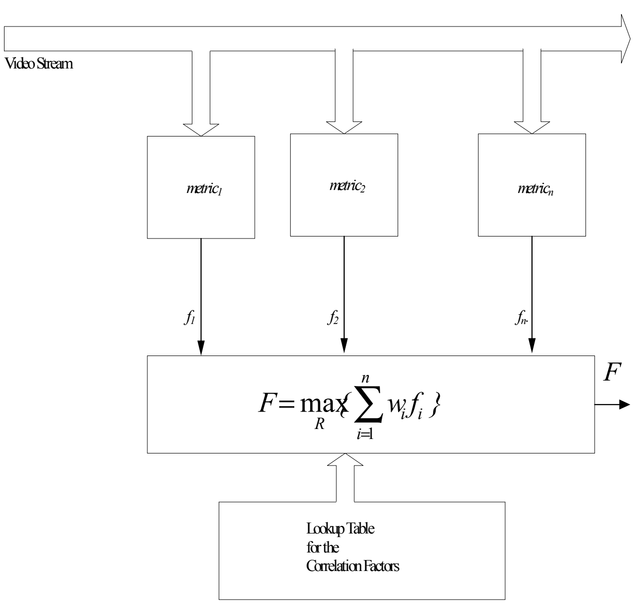

Figure 1 A schematic diagram for a scalable dynamic objective metric

Either the Pearson linear correlation factor or the Spearman rank order[13] can measure the correlation between the subjective and objective measures. The former assumes a linear relationship between the subjective and objective results, while the latter only assumes a monotonic relationship between them (could be linear or non-linear). We use the Spearman rank order (equation 2) to measure the correlation factor *R* between the subjective data set X_r and the objective data set Y_r.

$$R = 1 - \frac{6 * \sum (X_r - Y_r)^2}{n(n^2 - 1)} \qquad (2)$$

where the sum goes from *1* to *n* and *n* is the number of data samples in either set.

3. Applying Genetic Algorithms to Video System Design

A Genetic Algorithm (GA)[7] is based on a natural concept that diversity helps to ensure a population's survival under changing environmental conditions. They are simple and robust methods for optimization and search and have intrinsic parallelism. GAs are iterative procedures that maintain a population of candidate solutions encoded in the form of chromosome strings. The initial population can be selected heuristically or randomly. For each generation, each candidate is evaluated and is assigned a *fitness value*, which is the cost function as defined in section two. These candidates are selected for reproduction in the next generation based on their fitness values. The selected candidates are combined using the genetic recombination operation *crossover*. The *crossover* operator exchanges portions of bit strings to hopefully produce better candidates with higher fitness for the next generation. The *mutation* is then applied to perturb the bits of the chromosomes so as to guarantee that the probability of searching a particular subspace of the problem space is never zero.[3] It also prevents the algorithm from becoming trapped at local optima.[8,5] The whole population is evaluated again in the next generation and the process continues until it reaches the termination criterion. The termination criterion may be triggered by finding an acceptable approximate solution, reaching a specific number of generations, or until the solution converges.

We propose a flexible optimization paradigm. The optimization process utilizes GAs to come up with choices for the parameter settings, implementation alternatives and an interconnection scheme that achieve the best objective picture quality. In optimizing the video-processing scheme, a chromosome defines a certain way in which different video processing modules are connected and thus, the way video sequences are processed. A chromosome consists of a number of genes. The genes in the video optimization process are the video processing functions as well as their order, (which determines the connection scheme). Figure 2.a shows a general structure for the chromosome representing a video processing chain.

However, scaling the overall system design requires the ability to change the number of modules in a chain. Figure 2.a shows a chromosome representation for a video system, which consists of *n* cascaded modules. Should this process be scaled down, a smaller number of video processing modules is used instead. This is a typical situation in commercial products, when there are many lines of a certain product. Each line will have a certain degree of complexity, which is directly proportional to its performance. Figure 2.b and 2.c shows an exemplary case of two chromosome structures, which reflect the difference in two video systems. The former one has three video processing modules, and the latter has five modules.

functions' order gene	function_1 gene	. . .	function_n gene

Figure 2.a A general structure for the chromosome representing a video processing chain

functions' order gene	function_1	function_2	function_3

Figure 2.b A chromosome structure for representing a three-video-processing-modules video processing chain

functions' order gene	function_1	function_2	function_3	function_4	function_5

Figure 2.c A chromosome structure for representing a five-video-processing-modules video processing chain

Regardless the number of video modules comprising a video system, optimizing the system is carried out in the same fashion. Thus the proposed method allows for a scalable optimization of video systems.

4. Optimizing A Video Processing System

We optimized a video processing system, which consisted of four cascaded video processing modules, namely: a spatial poly-phase scalar, a noise reducer, a sharpness enhancer and a histogram modification module. The optimization algorithm deals with each module as generically as possible. It assumes no prior information about this module or its connectivity constraints (the casdaded modules' order). The optimization module perturbs each module's pre-defined set of parameters. The data precision (number of bits in a data bus, i.e., bus width) between two cascaded modules is considered a parameter to be optimized. We elected to use this set of video processing modules because of their vital role in any video system.[6] Moreover, some of these modules are competing modules,[12] e.g., increasing the sharpness would enhance the perceived existing noise and reducing the noise will blur the picture, resulting in the loss of its appealing crispness. The system consists mainly of the video processing system, the objective image quality measurement component and the genetic algorithm optimizer. The computational bottleneck in this scheme results from the complexity of the video processing system. We run a number of video processing systems in parallel (depending on the available processors on a parallel computer), as well as a number of the objective image quality metric components. This step of parallelizing the computationally greedy portions of the system enhances the performance significantly. Figure 3 shows a schematic diagram of the overall system.

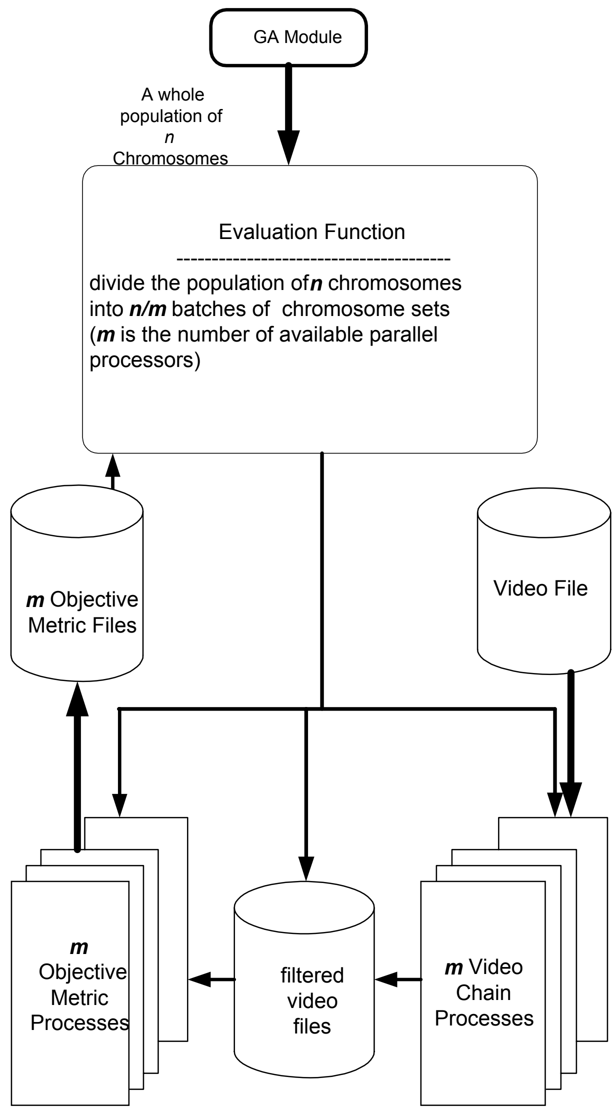

Figure 3 A schematic diagram for optimizing a video system

A detailed description of the video processing modules is introduced in section 4.1. Section 4.2 gives the details about the objective metric used and, finally, section 4.3 describes the optimization operation.

4.1. Cascading a Simplified Four-Module Video Processing Chain

The video-processing filter to be optimized comprises video-processing modules which are considered to be essential for high-end and top-end television sets. We deal primarily with video signals in the YUV and RGB domains, i.e., with image enhancement and display adaptation functions. Tuning, IF/color decoding, and channel/source decoding are not considered for now. The functions used are luminance peaking by sharpness enhancement, spatial scaling, noise reduction and histogram modification.

Sharpness enhancement, which nowadays is a fairly common feature in TV sets, focuses on improving the perceived sharpness of the luminance signal. Boosting the higher frequencies in the luminance signal basically enhances the sharpness. This may, however, lead to aliasing artifacts, which obviously needs to be prevented; that is achieved by a set of different sub-algorithms - contrast control, clipping prevention, dynamic range control and adaptive coring - which compete to reduce the aliasing artifacts. Each of them provides a gain factor that can safely boost the higher frequencies. A selector sub-unit decides which one of these competing gain factors will be used.

The noise reduction unit reduces the higher frequency components based on measuring the presence of noise, as will be explained in section 4.2.

The scalers are implemented using polyphase FIR filters. The *horizontal scalers* process each line of input video data and generate a horizontally scaled line of output video data. In the case of expansion, this is done by up-sampling that is performed either by a polyphase filter for which the horizontal expansion factor determines the filter phases required to generate each output pixel, or by a filter that uses this factor to interpolate the output pixels from the input pixels. In the case of compression, a transposed polyphase filter is used to down-sample the input data, and the horizontal compression factor determines the required filter phases. The *vertical scalers*, however, generate a different number of output video lines than were input to the module, with input and output lines having the same numbers of pixels. In the case of expansion, at least one line of video data is output for each line that is input to a polyphase filter, for which the vertical expansion factor determines the number of up-sampled lines generated in response to an input line, along with the required filter phases, or by a VPD filter that uses this factor to interpolate the output lines from the input line. In the case of compression, at most one line of video data is output for each line that is input to a transposed or non-transposed polyphase filter for which the vertical compression factor determines whether a down-sampled line is generated in response to an input line, along with the required filter phases.

Histogram modification stretches the luminance values for the black color and the white color to better represent the color content of the video sequence.

4.2. Building a Composite Objective Image Quality Metric

The video sequences could be contaminated with analog noise, which can be well approximated by white noise. However, the current growth of digital compression and multimedia video processing introduce a new set of artifacts, namely: blocking, ringing and mosquito artifacts. Among these, blocking artifacts are the most irritating to the human eye.[16] We are using a composite objective metric, which blends the evaluation of a white noise measurement unit, a blocking detector, a contrast-measuring unit and luminance signal clipping. The readings of these modules f_i, as in Figure 2, are weighted by weight factors w_i $1 < i < 4$, and are linearly combined. The weights are calculated to maximize the correlation (as in equation 2) between the overall objective image quality measure and the subjective evaluation on a pre-defined set of video sequences.

The noise measurement unit basically assumes variation in the flat areas of an image (low spatial frequency) is nothing but noise. To identify these areas, the image is divided into a number of small blocks and a measure of intensity variation is computed for every block. Assuming that the intensity of the noise is much smaller in magnitude than the signal, the block with least variation should correspond to a constant brightness region (described above). A high-pass filter or band-pass filter filters out the DC component and adds the filtered outputs to get a measure of the variance. The filters model the human visual perception characteristics and hence we get an estimate of the perceptually significant noise in the image. The output is clipped using a clipping function, which ensures that only the noise that contributes perceptually is counted. The clipping function thresholds are derived from Watson's model of perception threshold. He used this model to design a perceptually lossless quantization matrix for an image compression technique based on the Discrete Wavelet Transform (DWT). The model is described by equation 3

$$Y(f) = 10^{0.466(\log(f)+0.4)^2-0.31} \quad (3)$$

The blocking impairment metric (BIM) is primarily based on the measurement of intensity difference across block edges of the decoded image.[15,16] A rough measurement of the amount of blocking in a picture can be obtained by adding up the squared differences across the block boundaries of an image. The measurement is done separately for horizontal and vertical blocking. The metric for horizontal blocking (vertical edges) may be expressed mathematically as:

$$M_h = \|WD_c(f)\| = \sqrt{\sum_{i=1}^{N/8-1} \left\| w_i \left(f_{c(8i)} - f_{c(8i+1)} \right) \right\|^2} \quad (4)$$

where f is the image, D_c is the difference operator across columns, W is a weighting matrix defined according to the visual prominence of the blocking effect and w_i is the weight vector corresponding to the pixels of the image column f_c. The weights are computed as follows. For the difference of pixels at (i,j) and (i,j+1) the weight w_{ij} is defined as:

$$w_{ij} = \begin{cases} \dfrac{1.152*\ln\left(1+\dfrac{\sqrt{\mu_{ij}}}{1+\sigma_{ij}}\right)}{\ln\left(1+\dfrac{\sqrt{255-\mu_{ij}}}{1+\sigma_{ij}}\right)} & if \mu_{ij} <= 81.0 \\ & otherwise \end{cases} \quad (5)$$

where μ_{ij} is the mean of the 1-line strip of pixels on either side of the difference and σ_{ij} is their standard deviation. The factor μ_{ij} is a measure of the average brightness of the portion of the picture and hence takes care of the brightness-related property of human vision. The factor σ_{ij} is a measure of variation of intensity and is therefore used in the denominator of the weight. For the final metric, the above value is further normalized by the average inter-pixel variation inside the blocks. The normalizing factor E, is defined as:

$$E = \frac{1}{7}\sum_{k=1}^{7} M_h \quad (6)$$

The contrast-measuring unit averages out the difference between the maximum and the minimum luminance values per block in the image. The clipping unit simply counts the number of times the luminance value exceeds 95% of the maximum allowed value or falls below 5% of the minimum allowed luminance level.

4.3. Utilizing Genetic Algorithms to Optimize the Video Chain

The optimization algorithm uses a variant of the standard genetic search.[11] Here the initial population (*n* chromosomes) is generated randomly and each of the chromosomes is evaluated. An intermediate population is generated in the following fashion;

- The current population is copied to the intermediate population.
- Each chromosome in the current population is randomly paired with another chromosome and crossover is performed if the difference criterion is satisfied (see divergence below). The user can specify the crossover operator. The resulting children are evaluated and added to the intermediate population.

The resulting intermediate population has more than *n* chromosomes (2*n* if all the chromosomes pairs are different enough). The best *n* chromosomes from the intermediate population are selected and passed to the next generation. Note that no mutation is performed during this stage. Two chromosomes are crossed over only if the difference between them is above a threshold. This threshold is lowered when no chromosome pairs can be found with a difference above the threshold. When the threshold reaches 0, a re-initialization (*divergence*) of the population is done. Here the best chromosome available is selected as a representative and copied over to the next generation. Mutating a percentage (35 %) of the bits of this template chromosome generates the rest of the chromosomes.

The algorithm terminates when the number of divergences or failed divergences (those which did not improve the result) reaches a specified number. The user can also specify the maximum trials (evaluations) allowed, over all system structure and gene structure

5. Experiments

The video processing filter as proposed in section 4 has 60,000 possible variations; the noise reduction unit has a smearing factor of four settings (2 bits are needed to represent it), the sharpness enhancement has a parameter with five settings (3 bits are needed to represent it), the number of bits transferred between any two cascaded video processing units could range between 8 and 12 (5 settings; 3 bits are needed to represent it). Having 4 video processing modules, 24 possible ways of cascading them (4!) are possible (5 bits are needed to represent it). Thus, the chromosome needed to represent this video-processing filter comprises 19 bits.

The GAs improve the overall performance of a generation (a set of configurations for the video- processing filter, which resulted after crossover and mutation). Thus the average cost for each generation gets reduced. This is an indication that the GAs are pointing the solution in the right direction. Figure 4 shows the image quality improvement of the best video chain configuration over the path of evolution. The horizontal axis is the trial number over the whole path of evolution, which simply maps to the time of the optimization process, and the vertical axis is an absolute measure of the perceptual quality of the resulting image. The main goal of the optimization is to find a configuration with the *best* possible image quality. Thus, the best performance of a generation is the candidate for the global best configuration. Yet, it is being improved from one generation to the next. A stopping criterion is either hitting the hypothetical best or being unable to better improve the resulting image quality any further. After running almost 3800 video system configurations, the performance settled as shown in Figure 4. The resulting video sequences from the best configuration were examined as well as a number of random samples from the 3800 configurations and the results correlate highly with the subjective evaluation. Figures 6a – 6c show one frame from a processed video sequence, which resulted at different points in the evolution path. Figure 6a shows this frame at an early stage of the evolution process while Figure 6c shows the same frame when processed by a well-evolved video system, which resulted at the end of the evolution path. The quality improves for the resulting image, especially the picture's appealing crispness and the reduction of the noise.

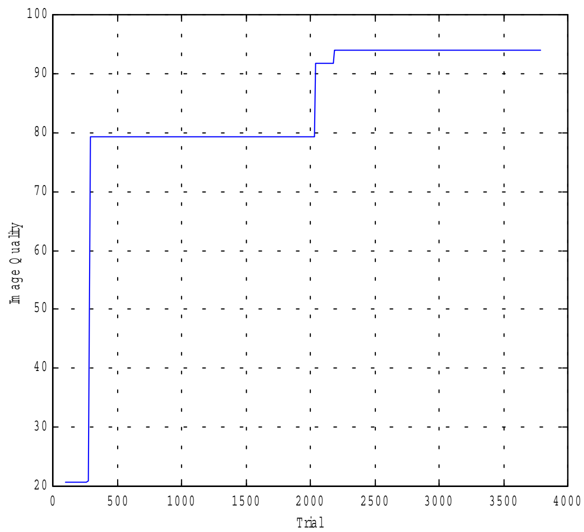

Figure 5 Best *performance over the evolution process*

Figure 6a The resulting picture at the beginning of the evolution process.

Figure 6b The resulting picture at the middle of the evolution process.

Figure 6c The resulting picture at the end of the evolution process.

6. Conclusions

In this paper, we presented our method for automatically optimizing a complicated video processing system, without any prior information about the constituent video processing components. We utilized a modified version of the genetic algorithm to improve its performance. Using an automatic optimization method necessitates the use of a cost function, which evaluates the perceptual image quality automatically. We introduced a method to combine a number of image quality metrics to maximize the correlation between the perceived quality and the measured objective quality.

7. Acknowledgements

The authors are deeply indebted to Dave Schaffer, Larry Eshelman and Keith Mathias from Philips Research Labs at Briarcliff, New York for supporting the genetic algorithm package and for their continuous support and technical help. We would also like to express our gratitude to Jorge Caviedes from VQEG and from Philips Research Labs at Briarcliff, NY for many useful discussions and feedback.

8. References

1. A J. Ahumada, Jr., " Computational image quality metrics: A review," in *SID Symposium, Digest,* vol. 24, 305-308, 1993.
2. C. J. Van den Branden and O. Verscheure, "Perceptual Quality Measure Using a Spatio -Temporal Model of Human Visual System," *Proc. SPIE*, vol. 2668, San Jose, CA, Jan. 1996, pp 450-61.
3. A. Chipperfield and P. Fleming, "Parallel Genetic Algorithms," in *Parallel and Distributed Computing Handbook* by A. Y. H. Zomaya, McGrawHill, 1996, pp. 1118-1143.

4. S. Daly, " The visible differences predictor: An algorithm for the assessment of image fidelity," in *Digital Images and Human Vision*, ed. A. B. Watson, 179-206, MIT Press, 1993.
5. D. E. Goldburg, *Genetic Algorithms in Search: Optimization and Machine Learning, Reading*, Mass. Addison-Wesley, 1989.
6. G. de Haan, *Video Processing for Multimedia Systems*, CIP-Data Koninklijke Bibliotheek, The Hague, ISBN 90-9014015-8, 2000.
7. J. H. Holland, *Adaptation in Natural and Artificial System*, University of Michigan Press, Ann Arbor, 1975.
8. P. Husbands, "Genetic Algorithms in Optimization and Adaptation," in book *Advances in Parallel Algorithms* by Kronsjo and Shumsheruddin ed., 1990, pp. 227-276.
9. ITU-R Recommendation 500-7, "Methodology for the subjective assessment of the quality of television pictures," ITU, Geneva, Switzerland, 1995.
10. ITU-T Study Group 12, Contribution COM 12-60, *An Evaluation of new methods for objective testing of video quality: objective test plan,* September 1998.
11. L. Eshelman, "The CHC Adaptive Seacrh Algorithm. How to Have Safe Search When Engaging in Nontraditional Genetic Recombination", in G. Rawlins, editor, *Foundations of Genetic Algorithms*, pp. 265- 283. Morgan Kaufmann, 1991.
12. A. Ojo, "An Algorithm for Integrated noise Reduction and Sharpness Enhancement", *Proceedings of International Conference on Consumer Electronics*, pp.58- 59, August 2000, Los Angeles.
13. L. L. Scharf, *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*, Addison Wesley Longman, ISBN: 0201190389, 1991.
14. VQEG, "Final Report From the Video Quality Experts Group on the Validation of Objective Models of Video Quality Assessment", March 2000. http://www.crc.ca/VQEG
15. Wu, H.R.,Yuen,M. "A survey of hybrid MC/DPCM/DCT video coding distortions" *Signal Processing Letters*, Vol. 70, pp. 247-278, 1998.
16. Yang, Galatsanos and Katsaggelos "Projection-Based Spatially-Adaptive Reconstruction of Block-Transform Compressed Images", IEEE Tans. on image processing, Vol. 4, pp. 896- 908, 1995.
17. W. Y. Zou, "Performance evaluation: From NTSC to digitally compressed video," *SMPTE journal* 103(12), pp. 795-800, 1994.