Mobile handsets have evolved from the bulky, heavy, voice-only communications systems of a decade or two ago to today’s ultra-compact devices equipped for Moving picture Expert Group-1 layer 3 (MP3) playback, still-and video image capture, high speed data transmission rivaling Digital Subscriber Line (DSL) services, Global Positioning Systems (GPS) and television services (Ping Wah 2005). Combined image acquisition, processing, storage, and communication capabilities in a compact, portable device make them an ideal platform for embedding computer vision and image processing capabilities in the pursuit of new mobile applications (Xu Liu 2008). The adoption of efficient standardized codecs such as Windows Media, H.264, and Moving Picture Expert Group Layer-2 (MPEG-2) will advance the capacity of embedded applications to incorporate and stream video. With a programmable platform, developers can not only select best codecs but also support a wide variety of codecs without code changes at the application level (Gene Frantz 2005). Use of appropriate codecs maximizes Intellectual Property (IP) reuse and reduces the cost reduction through
optimized hardware design. Advances in computational algorithms of image/video processing algorithms for mobile devices will increase overall video and imaging quality.

To bring these new computational algorithms on a mobile phone, it is necessary to reduce the complexity involved in video/image processing systems (Gene Frantz 2005). The key enabler to achieve these is an architecture that supports extensive design flexibility and programmability to deliver a quality video experience at acceptable system cost and portable size. The developers can support new multimedia standards and implement enhanced functionality to remain competitive and deliver efficient solutions to meet many consumer and industrial end equipment price/performance targets. The other solution is to use the optimized and efficient algorithms for the existing architecture and the adoption of standardized and efficient algorithms are needed to reduce the equipment price.

A.2 VIDEO/IMAGE PROCESSING ON MOBILE DEVICE

On March 10, 1876, Alexander Graham Bell first transmitted speech electronically, setting the development of telephone systems. Voice communication has remained as the primary function of telephone systems. Sharp launched its first camera phone product, the J-SH04, in November 2000, with a camera having 0.1M pixel resolution. Cameras have since become an integral “add on” to mobile phones. Based on a Gartner’s report, 48% of the mobile phones manufactured in 2006 were embedded with cameras, and this number already increases to 81% in 2010. The camera originally embedded as an accessory has the potential to alter significantly the usage of the mobile phone. Compared to traditional voice input on these devices, the camera input can gather more information with less effort in a single snapshot. As the proverb says, “A picture is worth a thousand words.” More importantly, the embedded camera has created the opportunity for the mobile user to interact with the physical world. The number and
variety of mobile devices has increased steadily in the last ten years. Ten years later (in 2010), some camera phones contain resolution up to 100 times the resolution of J-SH04 are introduced into the market. For example, the Samsung SCH-B600 has 10M pixel resolution and mainstream phones have at least 1 to 3M pixel resolution.

Figure A.1: Circuit board of a typical Ericsson cell phone (Courtesy: Sony Ericson)

Digital cameras are substituting the traditional film based devices, but most of the future planned growth is for digital cameras installed in cellular telephones. “Smart telephones”, which integrate several functions, such as MP3 player, digital camera, and Personal Digital Assistant (PDA) will slowly substitute the traditional voice-only systems (Dr. Margarita Esponda 2004). The cellular telephone will become a data centric and the hardware architecture will change accordingly. The Digital Signal Processors (DSP) is to cellular telephones what the microprocessor is to desktop systems, that is, the heart of the whole design. The figure A.1 shows the example of the circuit board of a typical Ericsson cell phone.

The DSP is the motor behind digital telephony. The first DSP was introduced in 1979, but the most successful design came from Texas Instruments (TI). DSPs from Texas Instruments have dominated the market for digital telephony, and today 65% of the
digital telephones uses chip made by TI (Dr. Margarita Esponda 2004). The 2G telephony systems are DSP centric, that is, their architecture is organized around a DSP chip with some memory and peripherals. The main advantage of DSPs is that they can be used as number crunchers, using limited frequency and power, due to the regularity of the computational load (mainly convolution operations for voice or signal processing).

A.3 HARDWARE DETAILS OF THE DSP

![Figure A.2 Block Diagram of TMS 320 DM 642 Processor (Courtesy: TI)]
The hardware features of the digital signal Processor DM 642 include 600/720 MHz, 4MB Flash, 32MB of 133 MHz SDRAM and 256 KB I2C EEPROM. For Video Capturing, processor contains 3 Female RCA connectors for composite video input (NTSC, PAL) and 1 Female S-Video connector for component (Y-C) video input (NTSC, PAL). For Video Display, processor has TVP5150/5146 Video Decoder, 3 Female RCA connectors (1 for composite video output, and 3 for RGB output or HDTV), 1 Female S-Video connector for RGB monitor output, 1 Female 15-pin VGA connector for RGB monitor output and On Screen Display (OSD) support FPGA. The Software features of the digital signal processor DM 642 include installation support for Code Composer Studio (CCS) v2.2 and v3.0. The figure 1.2 shows the block diagram of digital signal processor DM 642.
APPENDIX B

IMAGE SEGMENTATION & EDGE DETECTION

B.1 SCOPE

The chapter discusses about the implementation of image segmentation and edge detection algorithms for images. The different algorithms of image segmentation are implemented on a DSP to know the suitability of the algorithm for the biomedical applications. The different image segmentation algorithms are compared with respect to the computation time and memory requirement to fit the algorithm on a mobile device. The image segmentation using different edge detection operators are considered for the implementation and different edge detection operators are compared with respect to computation time and memory requirements.

B.2 INTRODUCTION

Image segmentation is widely used in all computer vision related applications and is the most prevalent technology in medical imaging. Image segmentation is the process of extracting features or regions of interest from an acquired image for further intelligent
computer analysis. The image is sliced into multiple regions based on characteristic properties such as intensity position texture position or some local or global statistical parameters. Segmentation using computer vision finds multiple applications especially in the area of biomedicine. The tissues or organs of interest need to be extracted from the images acquired using medical imaging techniques. The clear view of only the interested regions/organs will help the radiologist in earlier diagnosis and treatment. Segmentation is also used intensively in areas such as pattern recognition, remote sensing, metallurgy etc. There are number of literatures on image segmentation both semiautomatic and automatic.

**B.3 LITERATURE SURVEY**

The detailed survey reveals image segmentation and edge detection algorithms are reported. The watershed transform is widely used for image segmentation on computer vision applications. There are conventional as well as improvised segmentation algorithms depending on the application. The choice of the technique in most cases depends on the application and image in question rather than a generalized method (Shanthi K J 2009). A pixel-block scanning image segmentation VLSI based on a region-growing approach is discussed (Keita Okazaki 2008). Edge detection is a computer vision algorithm which is very processor intensive (Bellon, O.R.P. Silva (2002). It is possible to increase the speed of the algorithm by using hardware parallelism. An implementation of edge detection in an FPGA, the Altera NIOS2 development kit is presented (Keita Okazaki 2006). Details of the implementation, required hardware composition, implementation issues, and performance results of real-time real images are presented (Ikhlas Abdel Qader 2004) (Canny, J 1986).
B.4 METHODOLOGY

There are many algorithms for image segmentation. They are histogram thresholding, K Means clustering, region based techniques, watershed techniques, level set method, texture based techniques, edge detection techniques, wavelets based techniques, mean shift techniques, genetic algorithm based, JSEG based segmentation etc. Few algorithms such as histogram techniques, K Means, region grow, watershed, edge detection techniques are considered for comparison using computation complexity and memory requirements. Also the different edge detection operators such as Sobel, Prewitt, Robert cross, Canny and LOG are considered for the implementation. The image segmentation is considered for different wavelets such as Haar and Daub and the experiments are conducted.

B.5 IMPLEMENTATION DETAILS

The different biomedical images such as X ray, EEG, MRI and CT are considered for the experimentation and the different segmentation algorithms are compared. The sample of the result is displayed by considering the images such as cameraman image of TIFF format and size is 256 X 256, X-Ray of lungs of BMP format of size 673 X 743 and color image of fruits in JPEG format of size 297 X 233.

B.6 TEST RESULTS AND DISCUSSIONS

The experiment is conducted on several images of different formats, dimensions and different types. The biomedical signals considered for the experimentation and the sample result are shows in the figure B.1.
Figure B.1: Input images and image segmentation algorithms applied to images
GLOBAL THRESHOLDING

WATERSHED ALGORITHM

K-MEANS CLUSTERING (CLUSTER1)

Figure B.2: Image segmentation algorithms applied to images
The figure B.2 shows that the JSEG and watershed algorithms are better than the other algorithms. Histogram based methods are very efficient in terms of computation complexity compared to other image segmentation methods such as global thresholding and iterative thresholding which is reflected from the lower computational time shown in the table B.1. The iterative thresholding consumes comparatively more time compared to global thresholding because it takes time to compute the threshold iteratively in every step.
Table B.1: Computational requirements of few image segmentation algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Memory (KB)</th>
<th>Computation time (Seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Thresholding</td>
<td>6.429</td>
<td>0.15</td>
</tr>
<tr>
<td>Iterative Thresholding</td>
<td>6.672</td>
<td>0.179</td>
</tr>
<tr>
<td>Region growing techniques</td>
<td>9.652</td>
<td>0.217</td>
</tr>
<tr>
<td>Watershed algorithms</td>
<td>18.658</td>
<td>0.183</td>
</tr>
<tr>
<td>K-Means clustering</td>
<td>12.632</td>
<td>0.205</td>
</tr>
</tbody>
</table>

The table B.1 shows that if proper threshold is not fixed, it results in improper segmentation. For any size images, the global thresholding consumes approximately same time. Region-grow technique consumes more computational time as it considers a single pixel and eight connected components are compared for all the pixels until it reaches edge for entire region. It is found from the experiments, selection of the seed point and fixing the threshold are major points on which the output is dependent. The clustering algorithm is guaranteed to converge but it may not return optimal solution. The quality of the solution depends on the initial set of clusters and optimal value of k. An inappropriate choice of K yields very poor result. More time is consumed for calculating mean for several iterations and comparing each pixel with new mean till the algorithm achieves convergence. In watershed algorithms, the calculation of the gradient, comparison and growing process consumes more computational time.
Figure B.3: Image segmentation using edge detection operators
IMAGE SEGMENTATION USING LOG OPERATORS

IMAGE SEGMENTATION USING CANNY OPERATORS

Figure B.3 (Contd): Image segmentation using edge detection operators

Table B.2: Computational requirements of few edge detection operators

<table>
<thead>
<tr>
<th>Edge detection operator</th>
<th>Sobel</th>
<th>Prewitt</th>
<th>Roberts</th>
<th>LOG</th>
<th>Canny</th>
</tr>
</thead>
<tbody>
<tr>
<td>Memory</td>
<td>5.93</td>
<td>5.93</td>
<td>5.93</td>
<td>10.6</td>
<td>20</td>
</tr>
<tr>
<td>Computation time</td>
<td>0.16</td>
<td>0.16</td>
<td>0.16</td>
<td>0.54</td>
<td>0.3</td>
</tr>
</tbody>
</table>

From the table B.2, it is noted that for classical operators such as Sobel, Prewitt and Roberts cross, shifting operation has consumed maximum memory because operation
for gradients of both horizontal and vertical directions are taken in same function. For Laplacian Of Gaussian (LOG) operator, convolution operation consumes maximum memory because the convolution operation is done twice, one for Laplacian filter and Gaussian filter, and another for LOG filter and the image. For canny operator, convolution and thresholding consumes more memory, because this operator will do hysteresis thresholding.

B.7 SUMMARY & CONCLUSION

Image Segmentation is an important technique in the area of image processing with wide applications in medicine, remote sensing to mention a few. A lot of research work is in progress in various areas resulting in many computationally efficient algorithms. There are conventional as well as improvised segmentation algorithms depending on the application. The choice of the technique in most cases depends on the application and image in question rather than a generalized method. A real-time implementation of image segmentation and edge detection techniques on gray and color images image signals using different algorithms are presented. The algorithms are implemented using TI TMS320DM642 imaging developer kit. The results show that the execution time is low while the edge results are accurate.
APPENDIX C

Examiner 1 Evaluation Report - Ph.D. Thesis

1. Title is too long and can be better phrased - E.g. DSP implementation of novel algorithms for Video Processing on Mobile Devices
   Since the algorithms refer to those that run on Mobile Devices, it might be better to indicate that in the title.

   SOLN: Title is changed to “Digital Signal Processor implementation of some novel algorithms for Video Processing on mobile devices”

2. General Comments: Please double check the use of tenses properly. E.g. in page 4, it states "number will increase to 81% in 2010". 2010 is already over! Similar issue is there in 3rd paragraph of page 5.

   SOLN: The first draft was prepared in 2010. The correction is incorporated by adding New CISCO’S VNI report which is stated as follows. (Section 1.2, Page 2) According to CISCO Visual Networking Index (VNI) survey report, internet video consumption has increased significantly from an average of 700 terabytes/day in 2006 to 1200 terabytes/day in 2007. The video consumption has already expanded to 650% by 2011 to 7800 terabytes/day. Video content creation will expand even more rapidly in future days. According to the same market research firm, Internet video uploads grown from 500K uploads/day in 2007 to 4800K uploads/day in 2011.
Check grammar (entire thesis) before final binding. E.g. coding methods becomes … (should be "become") - Pg 3 Few more issues are there in the same page.

Singular/plural subject + verb usage: Please check grammar

SOLN: Correction is incorporated by changing the sentence. (Section 1.2, Page 2)
“Changed to coding methods become”

3. Abstract: One would expect a quick review of the research contribution in the abstract. It is not possible to see this in the first page of the abstract. Abstract should concisely summarize the problem and then provide the primary research contribution of the work.

Shorten the 2nd paragraph and bring the primary research contributions in to the first page of abstract. Since 4 areas have been highlighted, it would be better to re-write the abstract focusing on each area and the research contribution. This would have a greater impact on anyone reading the abstract.

SOLN: Abstract is reframed as follows

With advancement in technology, CISCO’S Visual Networking Index (VNI) forecast report has gained importance due to the fact that 4G services will lead to an explosion of new video applications that will transform telecom service providers as “experience providers.” The 4G services will make telecom providers to move into the ‘cable companies’ traditional territory by offering more comprehensive video services. According to CISCO’S VNI report (May 30, 2012), by 2016 annual global internet protocol is forecast to be 1.3 Zettabytes (10^{21} or 2^{70}). By 2016, 1.2 million video minutes, an equivalent of 833 days or over two years would travel the internet every second. Globally mobile video is projected to be the fastest growing consumer mobile service going from 271 million users in 2011 to
To provide comprehensive video processing services on a mobile phone, computational complexity and memory requirements of the video processing algorithms play an important role. H.264 is the open source advanced video coding (AVC) standard for Video coding/decoding and transmission. H.264/AVC standard is more economical with the use of bandwidth and the same has drawn the interest of telephone and data services providers. The video compression can be achieved using H.264 encoder and the encoded frame minimizes the memory requirements for storage and uses the available bandwidth efficiently for transmission. Some of the Video/Image processing services can be provided on a mobile phone using proposed research work are

- Video decompression to decode the encoded frames
- Video segmentation and summarization to minimize the memory requirements and to display the desired video frames
- Image compression of each frame by removing the redundancy
- Image scaling to display the decompressed image on any mobile phone with given resolution with acceptable quality.

H.264 decoder baseline profile is implemented on TI DSP and the experiments are conducted to decode the encoded frames for the input video. The experiments are conducted to identify the critical blocks in the decoder and the result shows that the deblocking filter consumes more than 30% of the computation time involved in decoding. The experiments are repeated for the decoder with and without deblocking filter and the experimental results shows that without deblocking filter it is possible to decode the encoded frames with acceptable quality of 34 dB. Also, it is possible to reduce the memory usage by implementing the decoder without deblocking filter.
It is observed that in specific applications, the decoded frames need to be segmented and summarized to minimize the memory space to store the video clips. This is particularly more useful for suiting applications such as news reading, cricket match, etc where the time constraints are important. The summarized frames should give the complete highlights of an event. The new technique or the method is proposed to segment and summarize the video to view the important frames. The methodology proposed is to modify the fitness function of the Genetic Algorithm named ad modified Genetic Algorithm and it is compared with Genetic Algorithm. The modified genetic algorithm gives better results for video summarization than the genetic algorithm.

The summarized frames of segmented video are considered to remove the redundancy to achieve image compression using pre-processing of singular value decomposition (SVD). Novel technique is proposed for preprocessing of SVD. In preprocessing, instead of computing the SVD for the entire image, SVD is applied to blocks of smaller dimensions. The entire image is divided into blocks of smaller dimensions such as 8, 16, 32, 64. The experiments are conducted for different block sizes and the experimental results show that a block size of 32 requires a computation time of 16 seconds to give the compression ratio of 2.62 in comparison with direct SVD which takes 24 seconds with a compression ratio of 2.12 to achieve a Peak Signal to Noise Ratio (PSNR) of 30 dB.

The decompressed image needs to be displayed on the mobile phone screen of any resolution. The scaling using interpolation can be used to scale the image to display the image on the screen using novel algorithm. The proposed algorithm replaces 2 X 2 block of the input image by 3 X 3 blocks. The new pixels of the replaced blocks are found by using simple procedure with less computational complexity. The experiments are conducted to measure the computation time and quality (PSNR) for the proposed method and comparison of the proposed method is
done with bilinear and nearest neighbor interpolation. The proposed scaling algorithm reduces the computation complexity compared to bilinear interpolation by utilizing a computation time of 0.018 seconds in comparison with bilinear interpolation which utilizes 0.856 seconds. The quality of the image is 34.5 dB which is better than nearest neighbor which gives a PSNR of 28.2dB.

The proposed research work has focused on these four aspects to achieve efficient video processing services on a given mobile phone by proposing novel techniques for each one of them. The proposed techniques (or algorithms) have been implemented on a hardware platform such as Texas Instruments Digital Signal Processor (TI DSP) to estimate the computational complexity in terms of time and memory requirements.

The exhaustive experiments are conducted for image segmentation and edge detection application and the comparison of the different methods are discussed in the appendix B.

The end of the first paragraph states "video traffic on a mobile phone will increase 4 folds between now and 2012". Thesis is dated March 2011. Does this mean within 9 months there will be an increase of 4 times? Or does "now" refer to the time research began?

SOLN: Correction is incorporated by adding the latest CISCO report. (Refer Abstract, Page iii)

According to CISCO’S VNI report (May 30, 2012), by 2016 annual global internet protocol is forecast to be 1.3 Zettabytes ($10^{21}$ or $2^{70}$). By 2016, 1.2 million video minutes, an equivalent of 833 days or over two years would travel the internet every second. Globally mobile video is projected to be the fastest growing consumer mobile service going from 271 million users in 2011 to 1.6 billion users in 2016.
4. **Acknowledgment:** Small typo - Postgraduate and Undergraduate are single words.

   SOLN: Correction is incorporated by changing the words to single words (Page vi)

5. **Chapter 1 - Introduction:** A Ph.D. thesis needs to start by clearly stating the motivation for the problem proposed to be addressed. Here one does not get any idea as to what the issues are by reading chapter 1 - too general. Problem statement appears in page 23. It is too far from the beginning. Some of the content covered in the introduction can be easily moved into an appendix without disturbing the flow.

   If the objective is to make the thesis self-contained, then generic literature about the area should be moved to appendix. The core part of the thesis should be left uncluttered so as to give the reader a quick impression of the problems addressed and solutions provided.

   SOLN: Introduction chapter is rewritten and some of the contents related to general discussions are moved to appendix. The flow of the Introduction topic is reorganized as follows.

   1. General information: Discussion on general information about the use of video/image processing algorithms on mobile phones
   2. Problem statement: Discussion about the problem statement of the research work
   3. Introduction to Image/Video processing: Discussion about the applications of image/video processing on mobile phones.
   4. Hardware implementation of the algorithms: Brief discussion about the hardware which can be used for implementation
   5. Image/video processing algorithms: Brief discussion about the H.264 advanced video coding standard, Video segmentation and summarization, Image compression and image scaling.
6. Survey of literature: Discussion about the literatures related to H.264, Video segmentation and summarization, image compression and image scaling.

7. Proposed solution: The section gives the proposed solution for the problems discussed in the problem statement.

8. Organization of the dissertation

Extended introduction topic in appendix A discusses introduction to mobile device, Video/Image Processing on mobile device and Hardware details about the DSP. (Page 116)

Some of the sentences do not make any sense: Pg. 2 - "Use of codecs maximizes IP reuse…" What does this mean?

SOLN: Intellectual Property (IP) reuse is required to reduce the design cost of System On Chip (SOC). Packaging of IP so that the blocks can be assembled easily to build a large system. The architectural exploration is required to understand cost, power and performance tradeoffs. (Appendix Page 116)

Pg 4 - "will increase to 81% in 2010" - Thesis submitted in March 2011. 2010 is past!

SOLN: The first draft is prepared in 2010. Hence the sentence is framed as will increase to 81% in 2010. The correction in incorporated by reframing the sentence is to “increased to 81% in 2010. (Appendix Page 117)

Page 5 - "65% of the digital telephones uses chip made by TI" - Please provide the source of this data.

Pg 9: Sec. 1.2 refers to "Image/Video Processing Algorithms". Then Sec. 1.2.1 suddenly talks about Video Decompression using H.264. Sec. 1.2 introduction should smoothly linked to 1.2.1 Please look at the cohesion of the text.

SOLN: Link is provided to provide continuity. (Section 1.5, Page 8)

Digital video compression techniques play a key role in recent multimedia communications. The limitation of bandwidth in communication channels and storage media demands more efficient video coding methods. On the other hand, introducing new applications and advances in multimedia technology demands video coding methods to include more complex and advanced features. Therefore, standardization of the video compression techniques is essential. Video compression (Video coding) is the process of compacting or condensing a digital video sequence into a smaller number of bits. ‘Raw’ or uncompressed digital video typically requires a large bit rate and compression is necessary for practical storage and transmission of digital video.

The computational algorithms which can be provided on a mobile phone include Video/Image compression, Video/Image segmentation, Video/Image scaling, edge detection, etc. The different computational algorithms considered provides non-core services in addition to standard voice calls which is benefit to both the subscriber and service provider.

Page 13: Image Compression Algorithm - In table 1.1. there are few issues that need to be addressed:

- Adjust font size/table column width to get the words properly fit
- Transmission time refers to the use of 28.8K modem - Do we use such modems any longer? One should make use of the basic speeds available in modern mobile handsets (3G, HSPA). Even GPRS (2G system) gives more than 56Kbps speed.
It is better if table 1.1 is redone using higher base speeds available with mobile phones in today's context.

SOLN: Correction is incorporated and the table C.1 is reframed as follows

Table C.1 Multimedia data types and uncompressed storage space, transmission bandwidth, and transmission time required.

<table>
<thead>
<tr>
<th>Multimedia Data</th>
<th>Size/Duration</th>
<th>Bits/Pixel or Bits/Sample</th>
<th>Uncompressed Size (B for bytes)</th>
<th>Transmission Bandwidth (b for bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A page of text</td>
<td>11” x 8.5”</td>
<td>Varying resolution</td>
<td>4-8 KB</td>
<td>32-64 Kb/page</td>
</tr>
<tr>
<td>Telephone quality</td>
<td>10 sec</td>
<td>8 bps</td>
<td>80 KB</td>
<td>64 Kb/sec</td>
</tr>
<tr>
<td>speech</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Grayscale Image</td>
<td>512 x 512</td>
<td>8 bpp</td>
<td>262 KB</td>
<td>2.1 Mb/image</td>
</tr>
<tr>
<td>Color Image</td>
<td>512 x 512</td>
<td>24 bpp</td>
<td>786 KB</td>
<td>6.29 Mb/image</td>
</tr>
<tr>
<td>Medical Image</td>
<td>2048 x 1680</td>
<td>12 bpp</td>
<td>5.16 MB</td>
<td>41.3 Mb/image</td>
</tr>
<tr>
<td>SHD Image</td>
<td>2048 x 2048</td>
<td>24 bpp</td>
<td>12.58 MB</td>
<td>100 Mb/image</td>
</tr>
<tr>
<td>Full-motion Video</td>
<td>640 x 480, 1 min (30 frames/sec)</td>
<td>24 bpp</td>
<td>1.66 GB</td>
<td>221 Mb/sec</td>
</tr>
</tbody>
</table>

Pg. 17: Given this is Ph.D. research work, I would expect a more mathematical oriented treatment of interpolation. Some of the qualitative aspects of interpolation are not properly mentioned. For example, a good interpolation technique needs to look at Geometric or Contrast invariance, preserving edges etc.,

SOLN: Correction is incorporated. The mathematical oriented treatment for interpolation is discussed in detail in the chapter 5. (Section 1.5.3, Page 13)

The corrected incorporation is as follows

Although it is difficult to compare methods and judge their output, the following basic criteria for good interpolation methods are discussed.
1. Geometric Invariance: The interpolation method should preserve the geometry and relative sizes of objects in an image. That is, the subject matter should not change under interpolation.

2. Contrast Invariance: The method should preserve the luminance values of objects in an image and the overall contrast of the image.

3. Noise: The method should not add noise or other artifacts to the image, such as ringing artifacts near the boundaries.

4. Edge Preservation: The method should preserve edges and boundaries, sharpening them where possible.

5. Aliasing: The method should not produce jagged or “staircase” edges.

6. Texture Preservation: The method should not blur or smooth textured regions.

7. Over-smoothing: The method should not produce undesirable piecewise constant or blocky regions.

8. Application Awareness: The method should produce results appropriate to the type of image and order of resolution. For example, the interpolated results should appear realistic for photographic images, but for medical images the results should have crisp edges and high contrast. If the interpolation is for general images, the method should be independent of the type of image.

9. Sensitivity to Parameters: The method should not be too sensitive to internal parameters that may vary from image to image.

Of course, these are qualitative and somewhat subjective criteria.

Pg. 23 - Problem Statement: Indicate the source (leading market research firm)

SOLN: Correction is incorporated by adding the source (CISCO VNI report) (Section 1.2, Page 2)
6. **Chapter 2 - Video Compression using H.264/Advanced Video Coding:**

   The objective of this chapter seems to do a comparison of different metrics on different video streams decoded on TI DSP. The benefit of such a comparison is questionable.

   SOLN: The main objective of the different algorithms stated in the problem statement is the implementation of the algorithms on mobile devices. Since the memory usage and computational complexity are the important issues for the implementation of the algorithms, critical routine of the different blocks of H.264 decoder are identified in the chapter 2. The deblocking filter block of H.264 decoder consumes almost 33% of the computation time. Also, the memory usage with deblocking filter block increases the power consumption. For the efficient implementation on a mobile device, deblocking filter is removed. The performance of the decoder with and without deblocking filter is measured in terms of different parameters. Such a comparison is required to justify the implementation of the decoder without deblocking filter. The quality obtained without deblocking filter is sufficient for the mobile phone applications. The result obtained encourages the implementation of the decoder without deblocking filter.

   **Author could have easily done a profiling job in an ordinary computer to identify which blocks is time consuming. Infact a comparison of profiling on a computer and DSP would have been more useful to evaluate DSP performance.**

   SOLN: The main objective is to identify the critical blocks which consumes more clock cycles in decoding the encoded frames. The objective is not to compare the performance of the DSP with computer. The optimized H.264 decoder code ported on DM642 DSP takes very less time for decoding compared to an ordinary computer.
I do not consider the work performed in this section (H.264 decoding) having a level demanded from a Ph.D. research.

7. Chapter 3 - Video Segmentation and Summarization:

The author straight away goes to using "Genetic Algorithm" for segmentation. The basis for this decision needs to be properly justified using quantitative and qualitative data. What are the reasons for excluding other algorithms such as Graphical models, Gaussian Mixture models, clustering etc.,

SOLN: The corrections are incorporated. (Section 3.2, Page 50)

The classification of the video segmentation algorithms includes pixel comparison, block based comparison, histogram comparison, feature based comparison, clustering based temporal segmentation and model driven video segmentation. The popular methods include uniform sampling and clustering.

The Genetic Algorithm (GA) is considered as it can be used for both video and image segmentation. The GA is simple to implement compared to Graphical models, Gaussian Mixture models, clustering model to implement on mobile phone. Memory usage, computational complexity and performance measure to segment and summarize are critical parameters for the selection of class of Video segmentation algorithm. GA is chosen as it gives a tradeoff between memory usage, computational complexity and performance measure.

- Uniform sampling is simplest method but suffer from undesirable repetition and poor access points for browsing compared to GA
- Clustering can produce good results but does not work with wide range of characterization that Genetic Segmentation Algorithm gets through evaluation function and does not support incremental segmentation.
Is the genetic algorithm based segmentation best for mobile devices? Pg. 49 para 3 indicates that GA based methods are more suited for segmentation and summarization. Citations are required to justify this statement.

Genetic algorithms are advantageous is that they are able to forego local optima in an attempt to reach the global optimum. This GA is less computationally expensive than exhaustive searcher and other adaptive techniques such as simulated annealing. GA cannot guarantee a global optimum but they usually give a good approximation through exhaustive search. This makes GA a good compromise between accuracy and computational complexity. The main drawback of GA is that they can take a long time to converge. The reason for using GA is its ability to deal with large complex search spaces in situations where only minimum knowledge is available about the objective functions. (Section 3.3, Page 51)


There is a bit of confusion in page 52 and 53. In pg 52, Figure 3.2 title says “Block Diagram …. Using Modified GA". Then pg 53 last sentence of 2nd Para states that "research work proposes to modify the given genetic algorithm shown in figure 3.2".

What exactly does figure 3.2 represent? Generic or the modified GA?

SOLN: Correction is incorporated. Figure shows the Block diagram of Video Segmentation and Summarization using Genetic Algorithm. Regarding the proposed method, the chapter 3 discusses in detail about the modification of the Genetic Algorithm by proposing a novel method. The proposed method is compared with GA and detail discussions are provided in the chapter 3. (Section 3.5, Page 54)
In equation 3.1 - how is α(i,j) determined? No information is given.

The GA fitness function is given by

\[ f(S_k) = \sum_{i,j \in S_k} \alpha(i,j) h(i,j) \]

Where \( \alpha(i,j) \) is a function for weighting the histogram differences \( h(i,j) \). Give less importance for images which are farther apart

\[ \alpha(i,j) = \begin{cases} 
\frac{1}{|i - j|} & ; i \neq j \\
0 & ; \text{else}
\end{cases} \]

Since the implementation of the algorithm is on a mobile phone, the complexity involved in computation is reduced by setting \( \alpha(i,j) = 1 \)

(Section 3.6, Page 57)

Pg. 58 says "Histogram gives an estimate of the difference between two frames". This is incorrect. Shouldn't this be "Histogram difference"?

SOLN: Correction is incorporated as Histogram difference. (Section 3.8, Page 60)

Frame # 81 in Figure 3.4 and Figure 3.5 are different!! This is the case for some other frames as well. I believe that frame numbers given are absolute with respect to the starting point and a given frame (number) must have the same objects. This does not seem to be the case in the two figures given 3.4 and 3.5. For example: Frame 81, 211, 251 and 261 in the two frames are different!!

SOLN: Correction is incorporated and the results are provided with correct frame numbers as shown in figures C.1 and C.2. (Section 3.8, Page 62-63)
Conclusion in page 62 is very subjective.

Overall conclusions for Video Segmentation is not very convincing. The profiling table (3.3) doesn't match with the conclusion given in pg 66.

SOLN: The conclusion part is rewritten by highlighting the result obtained using genetic algorithm and the proposed approach. (Page 63)

The proposed approach shows the summary of the event happened in only four frames $F_1, F_{31}, F_{101}, F_{261}$ as shown in the figure C.2. The other remaining frames $F_{81}, F_{211}, F_{221}, F_{251}$ are not significant to show the complete event as these frames
are almost same as the either frames $F_1$ or $F_3, F_{101}$ or $F_{261}$. Modified genetic algorithm explains the complete event with only our frames. Only 4 frames are sufficient to view the highlight of the video Hall Monitor. Hence the proposed approach gives the highlight of the complete event in only few frames. The results shown in the figures C.1 and C.2 indicate that modification in the genetic algorithm has helped to improve the efficiency of the GA for the input video clip. Some of the frames with almost similar information content are repeated in the GA and proposed approach removes those frames to a greater extent. The experimental results show that the proposed approach is better when memory is a constraint. Since the target is mobile applications where memory is a constraint and the proposed approach summarize only important frames. The summarized frames resulted from the proposed approach can be used view the highlight of the event occurred.

Table C.2: Profiling result for scaling factor of 10

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Genetic Algorithm</th>
<th>Proposed Approach</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Calls</td>
<td>Clock cycles</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td><strong>calc_cardinality_B</strong></td>
<td>1344</td>
<td>43.4</td>
</tr>
<tr>
<td>(calculate fitness function)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>calc_cardinality_C</strong></td>
<td>1344</td>
<td>44.8</td>
</tr>
<tr>
<td>(calculate fitness function)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>calc_fitness_func_v</strong></td>
<td>12</td>
<td>326.2</td>
</tr>
<tr>
<td>(calculate fitness function)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>calc_h_feature</strong></td>
<td>3</td>
<td>21</td>
</tr>
<tr>
<td>(to find histogram difference of two frames)</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>hist</strong></td>
<td>360</td>
<td>1047</td>
</tr>
<tr>
<td>(to find histogram matrix of the frames)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The profiling result obtained indicate that the number of clock cycles required to segment and summarize the frames using proposed approach is almost same as GA. The clock cycle 1 indicates the number of clock cycle spent including child function and clock cycle 2 is the number of clock cycles spent in a function excluding the clock cycle spent in its child functions. Hence, the profiling result in table C.2 indicates that the proposed method using modified GA doesn’t increases the computational complexity which is an important issue in mobile applications. The modified GA performs better than GA in segmentation and summarizing the frames without increasing the computational complexity. (Page 66).

8. Chapter 4 - Image Compression:

The work here addresses the issue of reducing the computational complexity associated with SVD for images. The flow of the content in this section is much better than those of previous chapters.

The results look good.

Pg. 86:  Sec. 4.6 - says "modified SVD can be modified …". This is not good writing. Please rephrase that sentence. Further improvements to modified SVD should be moved to a previous section. It should not be in the summary.

SOLN: Correction is incorporated. Sentence is reframed to “Also the modified SVD can be made adaptive to achieve higher compression ratio”. (Page 85)

For the improved MSVD, the experimental results are not given.

SOLN: For the improved MSVD, only the methodology is discussed.

For a $n \times n$ block with rank $R$ requires $2nR$ elements. If the block is split in to four quarters block of size $n/2 \times n/2$ with each sub block having rank $R_1, R_2, R_3, R4$ respectively, the number of storage element is $n(R_1+R_2+R3+R4)$
\[ n(R1 + R2 + R3 + R4) < 2nR \]  
(1)

Hence
\[ (R1 + R2 + R3 + R4) < 2R \]  
(2)

The above discussed method uses more computation time but the advantage of adaptive block size is that it can better map the region of complexity of the picture. The key to SVD compression is to use low rank approximation to the image. For the less complicated images, lower rank is sufficient to accurately represent it. For highly complicated images, higher rank is necessary for accurate representation. Hence, simple section can be represented with low rank approximation and complicated sections with higher rank approximation to include the detail.

The experiments are not conducted for the improved MSVD and hence the results are not provided. (Page 86)

9. Chapter 5 - Image Scaling:

I believe the assumption here is that after scaling 9 new pixels would map into the same CCD area occupied by 4 pixels under original resolution. If this is the case, then it is possible to calculate the contribution from the old to the new exactly. Why do we need to use different weight factors?

Example: New pixel N2 is formed by a combination of O1 and O2. If the exact overlapping is considered, it is possible to calculate the intersecting area of O1 and N2.

SOLN: The mapping is not to the same CCD area, but the research work mainly focuses on increasing the size of an image using image scaling with interpolation. The image scaling is required to display the video frame (QCIF) or an image on any display device of resolution such as CIF, QVGA, VGA, HDTV or D1.
In page 91, it states that new pixels N1, N3, N7 and N9 map into O1, O2, O3 and O4. This is not exactly the case. N1 maps into part of O1. So mapping is not 1:1.

SOLN: The mapping in the proposed algorithm is 1:1 for the pixels N1, N3, N7 and N9. The other unknown pixels such as N2, N4, N5, N6 and N8 are obtained by interpolation using proposed method.

What prevents the exact computation of Old Pixel contribution to the new pixels? My understanding is that weights can be exactly computed. I would appreciate if you can clarify whether my understanding is correct. If it is, the whole basis for this algorithm becomes very simple and very well deterministic. At the same time, the need for a1 to a6 disappears. Pg. 92 - There is no clear basis given for the values a1, a2, a5 and a6. A proper explanation is required.

What are b2 and b5 referred to in Pg. 96 under Sec. 5.5?

SOLN: (Section 5.3, Page 91)

Figure C.3: Novel method proposed for image scaling (Showing the block division operation for the original image)
Figure C.4: Novel method proposed for image scaling (Showing the block division operation for the scaled image)

The exact contribution of the old pixels to the new pixels is computed. Figures C.3 and C.4 explain the methodology proposed. Regarding the computation of the weights, the methodology is explained as follows

For an image of dimension m X n and block size $b_{l1} \times b_{l2}$, the number of blocks are $m/b_{l1}$ and $n/b_{l2}$. For a block size $b_{l1} \times b_{l2}$ in the original image and block size $B_{l1} \times B_{l2}$ in the scaled image after interpolation the number of substitution operations are $O_1 \times m/b_{l1} \times n/b_{l2}$, the number of multiplication operations are $O_2 \times m/b_{l1} \times n/b_{l2}$ and number of addition operations are $O_3 \times m/b_{l1} \times n/b_{l2}$ where $O_1$, $O_2$, $O_3$ are the number of substitution operation, multiplication operation and addition operation for each block. The different cases are considered for the proposed method to estimate the computational complexity in terms of number of substitution operation, multiplication and additions.
For Example,  
Let the dimension of the block size of the original image is $2 \times 2$ ($b_{l1} \times b_{l2}$)  
Let the dimension of the block size of the scaled image $3 \times 3$ ($B_{l1} \times B_{l2}$)  
Scaling factor is $(\frac{B_{l1}}{b_{l1}} \times \frac{B_{l2}}{b_{l2}}) = (1.5 \times 1.5)$

\[
O = \begin{bmatrix}
O_1 & O_2 \\
O_3 & O_4
\end{bmatrix}
\]
\[
N = \begin{bmatrix}
N_1 & N_2 & N_3 \\
N_4 & N_5 & N_6 \\
N_7 & N_8 & N_9
\end{bmatrix}
\]

Figure C.5: Illustration of block operation to obtain new pixels for interpolated image

Considering the $i^{th}$ pixel in the original image block $O$ as $O_i$ for $i=1$ to $4$ and $j^{th}$ pixels in the scaled image block $N$ as $N_j$ for $j=1$ to $4$. The edge pixel values of the scaled image block is same as that of original image block as shown in figure C.5 which is given by

$N_1$=$O_1$, $N_3$=$O_2$, $N_7$=$O_3$ and $N_9$=$O_4$  

(3)  
The remaining new pixels $N_2, N_4, N_5, N_6, N_8$ can be obtained from the pixels $O_i$ for $i=1$ to $4$ and the weight factor. Let $a_1, a_2, a_3$ and $a_4$ be the weight factors associated with the pixels $O_1, O_2, O_3, O_4$ respectively.

$\begin{align*}
a_1 \times O_1 + a_2 \times O_2 &= N_2 \\
a_1 \times O_1 + a_3 \times O_3 &= N_4 \\
0.5 \times (a_1 \times O_1 + a_2 \times O_2 + a_3 \times O_3 + a_4 \times O_4) &= N_5 \\
a_2 \times O_2 + a_4 \times O_4 &= N_6 \\
a_3 \times O_3 + a_4 \times O_4 &= N_8
\end{align*}$  

(4)  
(5)  
(6)  
(7)  
(8)  
The weight factors $a_1, a_2, a_3$ and $a_4$ are computed as follows.

$\begin{align*}
a_1 + a_2 &= 1 \\
a_1 + a_3 &= 1 \\
a_2 + a_4 &= 1 \\
a_3 + a_4 &= 1 \\
a_1 + a_2 + a_3 + a_4 &= 2
\end{align*}$  

Simplifying the equations we get $a_1 = a_4$ and $a_2 = a_3$  

Hence the equation (4) to (8) becomes
\[ a_1 \times O_1 + a_2 \times O_2 = N_2 \]  \hspace{1cm} (9)  
\[ a_1 \times O_1 + a_2 \times O_3 = N_4 \]  \hspace{1cm} (10)  
\[ a_2 \times O_2 + a_1 \times O_4 = N_6 \]  \hspace{1cm} (11)  
\[ a_2 \times O_3 + a_1 \times O_4 = N_8 \]  \hspace{1cm} (12)  
\[ 0.5 \times (a_1 \times O_1 + a_2 \times O_2 + a_2 \times O_3 + a_1 \times O_4) = N_5 = 0.5 \times (N_2 + N_8) \]  \hspace{1cm} (13)  

The proposed algorithm for scaling involves the mapping of a 2x2 block to 3x3 block. Why other possible mapping combinations are excluded is not very clear. Can we take a 3x3 as a base and then expand? What would happen if 3x3 blocks are taken with one row/column overlapping?

It would have been better had the author compared two such mapping combinations.

**SOLN:** (Page 99)

Table C.3: Computational complexity for different block sizes for the proposed approach (For each block)

<table>
<thead>
<tr>
<th>Block size</th>
<th>Block size</th>
<th>Substitution Operations</th>
<th>Multiplication Operations</th>
<th>Addition Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>b_{L1} X b_{L2}</td>
<td>B_{L1} X B_{L2}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2x2</td>
<td>3x3</td>
<td>4</td>
<td>9</td>
<td>5</td>
</tr>
<tr>
<td>3x3</td>
<td>5x5</td>
<td>9</td>
<td>44</td>
<td>24</td>
</tr>
<tr>
<td>2x2</td>
<td>4x4</td>
<td>9</td>
<td>13</td>
<td>7</td>
</tr>
</tbody>
</table>

Table C.4: Computational complexity for different block sizes for the proposed approach (For QCIF frame)

<table>
<thead>
<tr>
<th>Block size</th>
<th>Block size</th>
<th>Substitution Operations</th>
<th>Multiplication Operations</th>
<th>Addition Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>b_{L1} X b_{L2}</td>
<td>B_{L1} X B_{L2}</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2x2</td>
<td>3x3</td>
<td>25344</td>
<td>57024</td>
<td>31680</td>
</tr>
<tr>
<td>3x3</td>
<td>5x5</td>
<td>25488</td>
<td>124608</td>
<td>67968</td>
</tr>
<tr>
<td>2x2</td>
<td>4x4</td>
<td>57024</td>
<td>82368</td>
<td>44352</td>
</tr>
</tbody>
</table>
The computational complexity for the different block sizes are shown in the table C.3 and table C.4. From table C.3 and C.4, it is clear that the number of substitution operations, multiplication and addition operations required are less for block size 2 X 2 to 3 X 3 compared to the other blocks shown in the table C.3 and C.4. Hence the selection of 2 X 2 blocks in the original image reduces the computational overhead which is the key parameters for consideration in mobile applications. Block size 3 X 3 to 4 X 4 is not considered as it gives less scaling factor considered to block size 2 X 2 to 3 X 3 but it provides good quality. The block size 2 X 2 to 4 X 4 provides high scaling factor but it provides low quality. Hence it is not considered for the given mobile applications. The proposed approach for block selection is a tradeoff between the computation complexity, scaling factor and the quality in terms of PSNR and quality index. Hence a block size 2 X 2 to 3 X 3 is considered in the proposed method for image scaling. Also, the input considered is of QCIF resolution (176 X 144) and the block operation of block size of 16, 32 and 64 are considered in the previous chapter image compression to apply MSVD. Hence the block size of 2 X 2 to 3 X 3 is preferred as 2 X 2 divides the block size 16, 32 and 64.

The meaning of different weight combinations is not given. What is the difference between (0.4,0.6) and (0.3,0.7)? They obviously lead to different quantitative results but why? There should be more analysis than what is presented.

SOLN: The weight factor (0.4, 0.6) and (0.3, 0.7) indicates the values of (a₁, a₂) for contributing the pixels (O₁, O₂) and (O₄, O₃) for the block size 2 X 2 to 3 X 3. The factor (0.4, 0.6) indicates that the contribution of a₁ is 0.4 (the weight age given to O₁ and O₄) and contribution of a₂ is 0.6 (the weight age given to O₂ and O₃). Similarly the factor (0.3, 0.7) indicates that the contribution of a₁ is 0.3 and a₂ is 0.7. Hence both are different.
Can the author also come up with a name for this interpolation technique (after the above concerns are addressed)

SOLN: The methodology is named as “WEIGHTED AVERAGE INTERPOLATION METHOD”

10. Conclusions:

6.1 Principal Contributions: I do not think that any of the bullet points under H.264 decoder can be considered as contributions of this work. Results in Video Segmentation and Summarization are not very convincing.

"Contribution" section needs to be completely re-written. Contribution refers to a new methodology or new insight coming out of research. It is not a listing of the activities done. Please identify the "real" contributions of this research.

SOLN: Correction is incorporated and principal contribution is rewritten as follows. (Page 111)

In the research work on H.264 decoder, implementation of the decoder is ported on TIDSP to estimate the memory complexity and computational complexity. From the experimental results following points are noted. For an un-optimized version of the decoder, it is possible to decode less number of frames per seconds. The experiments are conducted for different set of encoded frames of different compression ratios. The results obtained from the different parameters such as MSE, PSNR, MSAD, SSIM and VQM indicates that it is possible to decode the encoded frames much faster without deblocking filter without sacrificing the quality of the video. Even though there is drop in the video quality, the methodology adapted by removing the deblocking filter can be adapted for mobile applications.
The critical routines of the decoder block are identified. It is clear that the deblocking filter consumes approximately 33% of the total time required for decoding. The memory usage and computation time can be minimized by implementing the H.264 decoder without deblocking filter. The proposed methodology is better in minimizing the power consumption by reducing the memory usage and decoding at a faster rate.

In the research work on Video segmentation and summarization, proposed method is considered for segmentation and summarization. The different videos of type fade-in and fade-out are considered for the experimentation and the following points are noted. The proposed approach of modified GA is discussed in detail and it is compared with GA. The proposed methodology for modifying the genetic algorithm involves the modification of fitness function. The fitness function of GA is modified by adding extra function which improves the result compared to GA. The experiments are conducted for video segmentation to segment and summarize the frames by using modified genetic algorithm. The experiments are conducted for different sampling factors to reduce the number of frames in a video to select only few frames to view the highlights of an event. The experiments are conducted for different number of segments for summarization.

- The experimental results show that modified GA works better than the original GA.
- The experiments are conducted for different video frames to estimate the computational complexity involved in modified GA in comparison with GA. The results show that modified GA consumes almost same time as GA.

In the research work on image compression, Experiments are conducted to estimate the memory complexity and code complexity involved in implementing
SVD in comparison with DCT and Haar transforms. The SVD with pre-processing is considered to reduce the computational complexity involved in SVD. Experiments are conducted to estimate the memory complexity and code complexity for proposed compression algorithm using modified SVD and compare the algorithm with the other existing compression algorithms. The different input images of different formats and different types are considered and the following points are listed.

- The proposed approach of pre-processing the SVD is discussed in detail to derive the relation between rank \( r \) of the block and rank \( R \) of the entire image.

- The modified SVD is considered for different block sizes such as 2, 4, 8, 16, 32 and 64 to estimate the computational complexity involved. The experimental results shows that the block size of 32 requires a computation time of 16 seconds to give the compression ratio of 3.12 for each block which is better than the original SVD for a rank of 33 to achieve a PSNR of 30 dB.

- The experiments are conducted for different set of input images to estimate the complexity involved in different functions. The critical functions are selected and optimization can be done to reduce the computation time.

In the research work on image scaling, a method is proposed to scale-up an image reduce the computational complexity involved in the proposed approach. The different input images of different formats and different types are considered and the following points are listed. The proposed approach of image scaling is discussed in detail and the approach is implemented on a DSP. The experiments are conducted to estimate the memory complexity and code complexity for proposed scaling algorithm and compare the algorithm with the other existing algorithms.
The proposed algorithm on scaling reduces the computation complexity in terms of clock cycles compared to bilinear interpolation by utilizing a computation time of 0.018 seconds in comparison with bilinear interpolation which utilizes 0.856 seconds. The quality of the image is 34.5 dB which is better than nearest neighbor which gives a PSNR of 28.2dB.

The proposed approach is considered in a noisy conditions and the approach is compared with nearest neighbor and bilinear interpolation methods for different types of noises.

11. Appendix 1: A 1.2 says "Image Segmentation is widely used … is the most prevalent technology in biomedical engineering". This should be corrected as "…. In medical imaging"

SOLN: Correction is incorporated as medical Imaging. (Page 121)

12. References:
1. Ref # 73 - mistake in the title of the paper (pg 125)

SOLN: correction is incorporated. (page 138)
1. While considering the image compression algorithms, why the research is considered only Haar, Wavelets and DCT for compression. Why the researcher chose wavelet transform?

SOLN: JPEG uses DCT, while JPEG-2000 will uses wavelet transform.

The DCT is chosen for the following reasons:

   a. DCT gives an approximate representation of DFT considering only the real part of the series.
   b. DCT is a fast transform with less computational complexity.
   c. DCT is more popular transform coding methods and widely used.

The wavelet is chosen for the following reasons:

1. Wavelets provide a basis set which represents data set in the form of differences and averages, called the high-pass or detail and low-pass or average coefficients, respectively. The number of data points to be averaged and the weights to be attached to each data point depends on the type of the wavelet.

2. Wavelets are the probing functions, which give optimal time-frequency localization of a given signal

3. Time complexity (Computational time) of many wavelet transforms can be calculated with O(N) operations. More general wavelets require O(Nlog₂N) calculations, same as that of DCT

4. It is possible to construct our own basis function, for wavelets, depending upon the application in hand. Thus, if a suitable basis function is designed for a given task, then it will capture most of the energy of the function with very
few coefficients. This freedom is curtailed in DCT, where one has only cosine functions as the basis set.

Haar is a primitive wavelet with less computational complexity and hence it is considered.

2. **What is the basis for choosing parameters for evaluation of performance of the algorithms? Are the parameters mentioned sufficient and standard?**

SOLN: The different parameters are selected for the comparison mainly based on objective and subjective measures. The objective measures for the comparison is done using the popular parameters used in Video/Image processing.

3. **Why Genetic algorithm is selected for optimization? Any other standard algorithms for optimization could have been considered.**

SOLN: The classification of the video segmentation algorithms includes pixel comparison, block based comparison, histogram comparison, feature based comparison, clustering based temporal segmentation and model driven video segmentation. The popular methods include uniform sampling and clustering.

The Genetic Algorithm (GA) is considered as it can be used for both video and image segmentation. The GA is simple to implement compared to Graphical models, Gaussian Mixture models, clustering model to implement on mobile phone. Memory usage, computational complexity and performance measure to segment and summarize are critical parameters for the selection of class of Video segmentation algorithm. GA is chosen as it gives a tradeoff between memory usage, computational complexity and performance measure.
- Uniform sampling is simplest method but suffer from undesirable repetition and poor access points for browsing compared to GA
- Clustering can produce good results but does not work with wide range of characterization that Genetic Segmentation Algorithm gets through evaluation function and does not support incremental segmentation.