Perceptual Video Quality Assessment Tool

BHARGAV POKALA and PAVAN BANDREDDY

Department of Electrical Engineering
School of Engineering
Blekinge Institute of Technology
37179 Karlskrona
Sweden
This thesis is submitted to the School of Engineering at the Blekinge Institute of Technology for the fulfillment of Master of Science in Electrical Engineering degree with emphasis of Telecommunications.

Contact Information

Authors:
Bhargav Pokala
email id: bhargav.pokala@gmail.com

Pavan Bandreddy
email id: pavan.bandreddy@gmail.com

University Advisors:
Prof. Hans-Jürgen Zepernick
School of Computing

PhD. Student Muhammad Shahid
School of Engineering

Tech.Lic. Andreas Rossholm
School of Engineering

University Examiner:
Dr. Benny Lövström.
School of Engineering

School of Engineering
Blekinge Institute of Technology
37179 Karlskrona Sweden
Abstract:

Compressing the video data is necessary for video transmission over networks with limited resources. Video quality at the end is based on the compression techniques used. In this research work, we mainly focused on compressing video files using H.264 encoding by considering some parameters and conducting a subjective test on the resulting compressed videos in a specially developed Perceptual Video Quality Assessment (PVQA) tool. This PVQA tool was developed as part of the thesis project. For the compression techniques, we have considered spatial resolutions (i.e. VGA and QVGA), temporal resolutions (i.e. 25fps and 12.5fps) and bit rate (i.e. 300kbps and 600kbps). By considering these parameters for three different test sequences, we down scaled and encoded to transmit over lossless channel. The received data is decoded and up scaled using open source software’s. For encoding and decoding of videos in H.264, we used JM reference software. For scaling the videos spatially and temporally, both FFMPEG and Virtual Dub are used. For up scaling the decoded video in spatial resolution (QVGA to VGA) two filter methods are used i.e. Bicubic Interpolation and Nearest Neighbor technique in Virtual Dub. For up scaling in temporal resolution (12.5fps to 25fps) the repeat frame method of Virtual Dub is used. These generated videos were evaluated by conducting subjective tests to determine the best technique that is suitable for transmission over a bit rate limited network. The developed PVQA tool satisfies the requirements of International Telecommunication Union (ITU) standard ITU-R BT.500-12. The PVQA tool will ask the subjects to grade the videos quality according to his/her experience and these grades are saved in excel sheets. The PVQA tool was developed under Java environment, with the VLC player embedded. The tool was designed to support the video. Results of the subjective video quality survey are summarized and finally conclusions are given.

Keywords:
Perceptual Video Quality test, Video Quality Assessment, Subjective test, ITU-T R BT.500-12, Encoding schemes, Video Compression, Motion Estimation, UMHexagon search.
Acknowledgments

We would like to specially thank Dr. Benny Lövström for his full time support in this research work. His encouragement during the work made us to reach our final goal.

We would like to thank Prof. Hans-Jürgen Zepernick, Mr. Muhammad Shahid and Tech.Lic. Andreas Rossholm for their all-time support and cooperation in research work.

We are greatly thankful to our parents for their pressure of support that helped us to reach our goal successfully. Finally, we would like to thank our friends and BTH staff for their support.

Bhargav Pokala and Pavan Bandreddy.
# Contents

Abstract iii  
Acknowledgements iv  
Contents v  
List of Figures vii  
List of Tables viii  

1. Introduction 1  
   1.1. Aim and Objectives 4  
   1.2. Research Questions 4  
   1.3. Expected Outcomes 5  
   1.4. Research Methodology 5  

2. Literature Review 6  
   2.1. Background 7  
   2.2. Quality Assessment 7  
      2.2.1. Objective Tests 7  
      2.2.2. Subjective Tests 7  
   2.3. Types of Evaluation Recommended by ITU-R BT.500-12 8  
      2.3.1. Environment Conditions for Conducting Survey 8  
      2.3.2. Test Methods Recommended by ITU-R BT.500-12 8  
   2.4. Test Methods Recommended by EBU Project Group 9  
   2.5. Related Work 10  

3. Generation of Video Sequences 13  
   3.1. Introduction for Video Generation 14  
   3.2. Process of Generating Test Video Sequences 14  
   3.3. Test Video Sequences Process 14  
      3.3.1. Conversion of Raw Video to Obtain Source Video 15  
      3.3.2. Normal Video 16  
      3.3.3. Down Scaling Process 16  
      3.3.4. Coding Process 16  
      3.3.5. Up Scaling Process 17  
      3.3.6. I Frame Effect 18  
      3.3.7. Test Videos Sequence 18  

4. Design of the PVQA Tool 21  
   4.1. Architecture Design of PVQA Tool 22  
      4.1.1. JRE System Library 22  
      4.1.2. Referenced Libraries Package 23  
         4.1.2.1. Java Native Access 23  
         4.1.2.2. Platform 23
# List of Figures:

3.3. Test video sequences considered in our work .................................................. 15
3.3. Overview of video generation process for subjective test .............................. 19
3.3. Parkrun Sample videos generated for 300kbps and 600kbps ....................... 20
4.1. Jar files embedded for the development of PVQA tool ............................... 23
4.2. Outline representing architecture arrangement of PVQA tool ..................... 25
4.3. Location of JDBC driver required for Database operation ......................... 26
4.3. Database Design in Mostool ................................................................. 27
4.4. Initial screen created after database creation, when software is initiated .... 32
4.4. Admin Login Screen, generated after selecting admin mode ...................... 33
4.4. Admin settings form in new survey panel ............................................... 35
4.4. Pop-up window after saving settings and before finalizing to conduct the survey 38
4.5. Subject details screen before accepting security policy button .................... 39
4.5. Subject details screen after accepting security policy button ...................... 40
4.5. Subject viewing reference video at T1 duration ......................................... 41
4.5. Subject viewing Non-reference video during T1 session ............................ 42
4.5. GUI of T2 session, when subject mean to grade the test sequence ............ 43
4.5. GUI of survey session when all the test sequences are graded .................. 44
4.6. Design model of PVQA tool ..................................................................... 45
4.6. Use case in Admin mode .......................................................................... 46
4.6. Use case in Subject mode ......................................................................... 47
4.6. Flow chart of initial mode in PVQA tool .................................................. 49
4.6. Flow chart of Admin mode process in PVQA tool .................................... 50
4.6. Flow chart of Subject mode process in PVQA tool ................................... 56
5.2. System configurations for PVQA test ......................................................... 62
5.2. PVQA survey instructions giving to the subjects ....................................... 64
5.3. Video sequence flow in PVQA tool ............................................................ 65
6.2. Average scores and 95% CI for video sequences generated for 300kbps .... 70
6.2. Average scores and 95% CI for video sequences generated for 600kbps .......... 70
6.2. Subjective quality of video formats coded at 300kbps compared to normal video format 73
6.2. Statistical analysis for quality of video formats coded at 600kbps .......... 74
List of Table:

2.3. General viewing conditions for subjective assessment in laboratory environment 8
2.6. Grading scales in ITU-R BT.500-12 and SAMVIQ 10
3.3. JM software-coding parameters 17
4.3. Admin table 28
4.3. User table 28
4.3. Video table 28
4.3. Video Grading table 29
4.3. Custom Scale table 29
4.3. Binary Scale table 30
4.3. Global Properties table 30
4.3. Survey table 31
4.3. Timer Settings table 31
4.5. ITU-R BT.500-12 five point scaling 36
4.5. Types of scales available for subjective grading 36
4.5. Timer settings for PVQA test 37
5.2. List of parameters considered from ITU-R and SAMVIQ recommendations 63
5.2. Experimental setup measurements 63
6.2 Statistics of per video with average score and Confidence interval 95% 69
6.2 Statistics of proposed formatting technique with confidence interval 95% 71
6.2 Average and CI of 95% for each formatting technique with respect to 300kbps bit rate test sequences 71
6.2 Average and CI of 95% for each formatting technique with respect to 600kbps bit rate test sequences 71
6.2. Video quality level of scaled video formats compared to normal video format generated for 300kbps 72
6.2. Video quality level of scaled video formats compared to normal video format generated for 600kbps 72
INTRODUCTION
Chapter 1

1. Introduction

In this mobile internet era, huge amount of multimedia transmission over networks results in conjunction and distortion for internet services. One of the major types of multimedia transmitted over networks is video data, which plays a vital role in communication and entertainment. Video transfer over network resources is facing many network related issues like delays frames freezing etc: which results in quality degradation. Video quality metrics and quality measurement tools are necessary for the researchers to identify the degradation in video quality at the end user. Video Quality Experts Group (VQEG) and Moving Pictures Experts Group (MPEG) are two main researcher teams on video quality. These two groups joined to form team named Joint Video Team (JVT) for the development of video quality standards, metrics and specifications. Transmission of huge amount of video data over network leads to traffic delay and congestion. In order to overcome the impact of video transmission over network constrains, compression technique is used. Compressing the video results in reduced in video data size, which makes feasible to transfer over networks within limited resource. At the receiver, video is decompressed to its original size for obtaining the transmitted source video. Depending on the compression techniques of video transmitted over error prone network may degrade the video quality [15]. Quality of the video that was received over a network can be measured by performing subjective test and/or objective tests. Objective test results are based on mathematical calculations and are not as reliable as the subjective tests. However, the subjective tests are more time consuming and costly, we implemented subjective test MOS for measuring video quality in our thesis work. This work can then be the basis for developing or refining objective methods.

In order to conduct this subjective test in a convenient way, a Perceptive Video Quality Assessment Tool is developed. PVQA tool is a video quality metric based on subjective tests and running the GUI's recommended for the video quality evaluation. In this thesis work, we have considered three standard raw videos namely “Parkrun”, “Ducks Take Off” and “Mobile Calendar”. These raw videos are processed to reduce the data size for transmission over a channel. The encoding process depends on the type of transmission using and its control rate, we have considered baseline configuration for two different bit rates during encoding which is feasible for mobile transmission. We scaled the raw video sequences to spatial and temporal resolutions i.e. frame size and frame rate respectively. By scaling and coding these raw videos, we generated 24 test sequences i.e. eight test sequences for each scene. In addition to obtain précised results for subjective tests, we considered “Shields” as a training video for our survey and generated four test sequences similar to the above test sequences generation process with single bit rate. These training videos have been prioritized to play first during each survey and grading of these videos is not considered for our survey results.
PVQA is a subjective quality measurement metric with Mean Opinion Score (MOS) as a major grading scale. The PVQA tool was developed according to ITU standards ITU-R BT.500-12 quality assessment recommendations for subjective tests. According to ITU-R [1], video quality measurements are estimated with both objective quality measurement i.e. mathematical model and subjective quality measurements i.e. Human Visual System (HVS). Subjective video quality is widely accepted as a measurement technique for accurate results to assess the subject response. In this, subjective quality experiments; subjects are shown a series of videos either in sequence or in shuffle mode, to grade the video according to his/her Quality of Experience (QoE). Subject can grade the test sequence when the video is stopped. PVQA tool is developed with grading scale, playing time, grading time and pause time that are recommended by ITU. In addition, more features are added facilitate utilizing the tool for research.

To develop the PVQA tool and conduct the subjective video quality assessment test in professional manner, the rules and specifications of ITU-R BT.500-12 are considered. This ITU-R BT.500-12 [1] includes the test environment, methodology, test video sequence, number of test sequences and subjects (assessors), video length, grading time and pause time, grade scale, lab/home environment, etc. According to ITU-R BT.500-12, there are different types of video quality measurements methods like Double-Stimulus Impairment Scale (DSIS), Double-Stimulus Continuous Quality-Scale (DSCQS), Single-Stimulus and Stimulus-Comparison methods; Single Stimulus Continuous Quality Evaluation (SSCQE), Simultaneous Double Stimulus for Continuous Evaluation (SDSCE), Multidimensional Scaling methods and Multivariate methods. Our project mainly focuses on SSCQE methodology.

The overall work flow of this report is literature study to understand the concept of perceptual video quality tests and study of ITU recommendations that guides for conducting subjective tests will be observed in Chapter 2. Chapter 3 discuss about generation of test video sequence procedure. Process and performance for the development and functions of PVQA tool are discussed in Chapter 4. Chapter 5 provides the implementation of generated video formats into the PVQA tool and conduction of subjective tests i.e. experimental setup. Finally, analysis of the video quality based on survey results and conclusions will be shown in Chapter 6.
1.1. Aim and Objectives

The aim of this thesis work is to develop software for testing the subjective perceived quality of video sequences and to perform such tests. This work also includes identifying best video processing technique out of four possibilities for specified bit rate limitations, by using the PVQA tool. For this survey, we encode/decode the raw video sequences with respect to distinct frame size, frame rate and bit rate over a channel. For generating these video formats FFMPEG, Virtual Dub and JM reference were used.

Objectives of this research work are:

- Study of subjective video quality measurement requirements.
- Study of tools required for coding different video formats.
- Literature review on video quality matrices and evaluation tools.
- Generate survey required videos based on parameters: frame rate, frame size and bit rate.
- Develop a PVQA tool in Java IDE environments Eclipse and Net Beans.
- Generated test videos will be used for conducting survey in the developed PVQA tool.
- Conclude results and suggestions based on survey.

1.2. Research Questions

- From the proposed video formats, which video format is best suited for channel transmission over limited bit rate 300kbps and 600kbps?
- What is the purpose of proposed filtering or downscaling and up scaling of the raw video for network transmission?
- How does this PVQA tool support the researcher in conducting subjective quality test?
1.3. Expected Outcomes

✓ According to the subjective grading of the test video sequences, the best video format will be defined.

✓ A well-featured PVQA tool for conducting subjective video quality metric.

✓ Results in excel sheet and graphs are plotted for further analysis and conclusion.

1.4. Research Methodology

For conducting video quality tests, a subjective quality metric PVQA tool was developed in Java environment. A comprehensive study of subjective tests from ITU-R standards will be considered as a requirement for formulating and developing this tool. To develop the PVQA tool under Java environment Net beans and Eclipse IDE’s are used. For generating test sequences based on frame rate, frame size and bitrates; open source software’s FFMPEG, Virtual Dub and JM reference software are used. Reducing the videos spatially or temporal resolutions will reduce the file size, which makes ease to encode and transfer over network limitations. The received video files will be decoded and up scaled to its original spatial or temporal resolution and survey is conducted to identify best format.

The developed PVQA tool provides simple GUI for subjective tests and provides conducted survey results simultaneously. The induced test sequences are evaluated based on subjective test conducted in PVQA tool for different age groups, skill levels, genders and country of origin. The developed PVQA tool for subjective tests on video sequences can be updated with libraries packages that will give support to conduct the subjective test for audio and image sequences. The following chapter will give details of related works.
LITERATURE REVIEW
Chapter 2

2. Literature Review

2.1. Background

This chapter will provide the background, concept of perceptual tests and perceptual quality. Basics of quality assessment by subjective tests and objective tests are stated in section 2.2. In section 2.3 the types of subjective test that were recommended by ITU-R standards are stated. EBU recommendations for subjective test are given in section 2.4 and section 2.5 will give related research works done on perceptual test conducted on video quality.

2.2. Quality Assessment

In order to measure the changes in the quality of a video we have two methods: Objective Quality assessment and Subjective Quality assessment.

2.2.1. Objective Tests

Objective quality metric is a video quality metric [3] the quality is calculated using a mathematical formula by considering the video parameters like Signal to Noise Ratio (SNR), Bit Error Rate (BER), Peak Signal to Noise Ratio (PSNR), Quantization Parameters (QP) and Mean Squared Error (MSE). Depending on these calculations, the quality level of a video is stated. For instance, an interesting method of evaluating the video quality based on decoder parameter extraction was proposed in [2].

2.2.2. Subjective Tests

Subjective evaluation is a method of video quality metric where [3] the quality of a video will be evaluated by conducting a survey with the test sequences and grading them on the scale. Here each subject (assessor) will grade the video according to his/her perception level of quality; finally a Mean Opinion Score (MOS) is summarized from survey results. Subjective tests were based on the HVS, where the subject will grade the quality according to his/her own perception. These grading values will differ from subject to subject due to age, gender, country of origin, skill level, etc. In order to conduct a subjective test, the ITU-R recommendations for subjective assessment of the quality of television pictures [1] are considered. Subjective tests are more time consuming and expensive but the result of these tests is considered as standard or true value. Objective tests are not fully correlated with subjective test results [14]. In our
thesis work, the generated video formats were evaluated by conducting subjective test in the developed PVQA tool.

2.3. Types of Evaluation Recommended by ITU-R BT.500-12

This section will give the general recommendations stated by ITU-R BT.500-12. This recommendation is followed for conducting subjective test in a professional way and includes general methods of test, grading scales and viewing conditions.

2.3.1. Environment Conditions for Conducting Survey

According to ITU-R BT.500-12 Standards, Subjective assessment for video quality can be kept under two environments: Laboratory viewing and Home viewing environment. Table 1 shows the requirements for conducting the test in laboratory environment.

<table>
<thead>
<tr>
<th></th>
<th>Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Ratio of Luminance of inactive screen to peak luminance: ≤0.02</td>
</tr>
<tr>
<td>2</td>
<td>Ratio of Luminance of the screen, when displaying only black level in a completely dark room, to that corresponding to peak white: ~0.01</td>
</tr>
<tr>
<td>3</td>
<td>Ratio of Luminance of background behind picture monitor to peak luminance of picture: ~0.15</td>
</tr>
<tr>
<td>4</td>
<td>Chromaticity of background</td>
</tr>
<tr>
<td>5</td>
<td>Other room illumination:</td>
</tr>
</tbody>
</table>

Table 1 General viewing condition for subjective assessments in laboratory environment [1]

In our thesis work, we intend to conduct the subjective tests by considering laboratory environment conditions at our university BTH, Sweden.

2.3.2. Test Methods Recommended by ITU-R BT.500-12

According to ITU-R, recommendations there are several methods for conducting subjective tests.

- Double Stimulus Impairment Scale (DSIS).
- Double Stimulus Continuous Quality Scale (DSCQS).
- Single Stimulus (SS), Stimulus Comparison Quality Evaluation (SSCQE).
  - Single Stimulus Continuous Quality Evaluation (SCCQE).
  - Simultaneous Double Stimulus for Continuous Evaluation (SDSCE).

Each subjective test methods assessment process is explained in detail in [1]. Out of these, we considered SSCQE as more preferable method for our
subjective test. SSCQE method is used for video quality measurement for continuous quality evaluation. Here the subjective is able to view the sequence once and according to his/her perception; he/she has to grade on the scale within the grade time (e.g. 10 sec). It states that, the overall test session should be no longer than 30 min and the sequences have to play in pseudo random order. By considering all these properties of SSCQE we have implemented this screening method in our work. Based on this we have developed our PVQA tool that is able to play set of different scenes and shuffle within the set of same scene with respect to time.

2.4. Test Methods Recommended by EBU Project Group

As we mentioned about test methods of ITU-R standards in section 2.3.2, there are few more methods rather than SSCQE, which performs different ways of conducting subjective tests like presence of hidden reference, display type, etc. In order to conduct subjective video quality tests for video codec’s, video formats, packet loss, freezing, etc EBU project group has developed new method called Subjective Assessment Methodology for Video Quality (SAMVIQ). Initially this method is developed for the evaluation of video codec’s in EBU’s research work. Where they have considered four different codec’s that are designed for Internet use like Windows Media 9, Real Networks 9, Quick Time 6, and MPEG-4. Finally, they have a good response from the survey with respect to the methodology used. Subjective methodologies is more time consuming, more costly and requires more efforts than other methods like mathematical models, etc. The results of these subjective tests are accurate, since the results are based on the Human perception levels which are the ultimate requirement for the research to improve the QoE. Objective tests use mathematical models for the analysis of video quality, which differ from the subjective tests analysis. According to ITU-R standards the process of conducting the subjective test are considered on a Display type TV rather than a PDA or PC. The difference of conducting subjective tests in display type TV and PC in detail referred in [17]. This is one of the issues for not considering the complete methodology of SSCQE recommended by ITU-R in our subjective test. Since our research work related to video quality over mobile networks, in our subjective test we intend to conduct on display type PC. One main feature of conducting test based on SAMVIQ methodology by considering the hidden reference can improve the subjective results efficient when compared with SSCQE methodology without hidden reference. The grading values of hidden reference are not considered for the conclusion of results.

According to the test organization recommended by SAMVIQ method [17], the test session with different scenes should follow same spatial format i.e. either VGA or QVGA. The process of randomizing will improve the assessment level from subject to subject. The length of all test sequences should be maintained static. As the ITU-R BT.500-12 grading scale refers to 5-point scale, which has limited grading levels. For
obtaining accurate level of grading, SAMVIQ’s 0 to 100 point scale is preferable. This 0 to 100 point scale also represented as ITU-R BT.500-12’s 5-point scale shown in table 2.

<table>
<thead>
<tr>
<th>Grading Scale</th>
<th>ITU-R BT.500-12</th>
<th>SAMVIQ</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bad</td>
<td>0 to 20 points</td>
</tr>
<tr>
<td>2</td>
<td>Poor</td>
<td>20 to 40 points</td>
</tr>
<tr>
<td>3</td>
<td>Fair</td>
<td>40 to 60 points</td>
</tr>
<tr>
<td>4</td>
<td>Good</td>
<td>60 to 80 points</td>
</tr>
<tr>
<td>5</td>
<td>Excellent</td>
<td>80 to 100 points</td>
</tr>
</tbody>
</table>

Table 2 Grading scales in ITU-R BT.500-12 and SAMVIQ.

The combination of SAMVIQ methodology and SSCQE methodology of ITU-R at some points may result in subjective test results, which are more preferable. So, we tend to consider both methodologies into one method and conduct the subjective survey for our work.

2.5. Related Work

This section will give details on related research work done on video generation and quality assessment; comparing with our proposed method of video formats transmission, subjective tests methodologies and subjective tools, etc.

Digital transmission of a video over network will be done in bit stream format with any compression, and the quality of video depends on the bandwidth limitations. For archiving better quality of video certain parameters are to be considered like; frame rate, frame size, quantization parameter etc. Based on bandwidth limit of the channel, the video can be compressed and encoded, which may affect both temporal resolution and spatial resolutions. After the encoding process, the video is transferred in bit stream format over channel, to attain better quality of experience higher bit rate is preferred. Therefore, the impact of these parameters will vary the quality of video. In paper [6] and [7] they have conducted Perceptual video quality assessment in two different ways, one is by considering both spatial resolution and quantization artifacts and other is based on frame rate (temporal resolution) and quantization artifacts. For our research work, both spatial resolution and temporal resolution of a video are considered before the encoding.

In the paper [4], when a raw or a reference-uncompressed frame of a video compressed to a bit rate of 0.5 Mbps and decoded with the MPEG-2 format were not similar. Where the quality of the reference and the decoded frame are varying much as shown in figure 1 of paper [4], the piano, hands of the harp player, stripes on the shirts and total frame is blurred. For digital video system the SS and TS content of source video plays a vital role to define the range of compression that can be possible. Similarly, for attaining better quality of video different codec’s and bitrates [8] are to
be considered. In our research a standard H.264 bit stream format is used, which is most commonly used formats for compressing high definition videos over internet for mobile transmission. Coming to the point of rate control, according to [4] HVS are more sensitive in compression than, reduce in spatial or temporal resolutions. In the case of specific compression rate the viewers prefer high resolution for both spatial and temporal, during encoding session the terms; compression rate, spatial and temporal resolutions plays a vital role and they are depended on the source frame size, frame rate and target bit rate. For our test video generation we considered the rate control parameters with two target bit rates 300kbps and 600kbps.

The technique of motion estimation embedded in H.264 standard video coding is essential to achieve a good compression. The motion estimation of a video is one major part of the coding; this is a part where the H.264 codec spends a large part of its calculations [9]. In order to reduce this motion estimation time there were many researches proposed many algorithms that improve the coding efficiency. One of the well know standard coding tool JM reference software assembled with UMHexagon search algorithm uses Dynamic Search Range (DSR) algorithm which speeds up the decision process when compared to other search algorithms[16]. This algorithm uses motion vector information of the neighboring coded blocks for search strategy optimization, with negligible PSNR loss and higher bit rate. In order to save the coding time and maintain the performance of coding, for our work the original JM reference software Motion Estimation Control Parameters i.e. UMHexagon search algorithm with search range prediction (DSR algorithm) is used. In [10] UMHexagon search mode provides good rate distortion performance with high video quality situation with no local optimum.

Video can be encoded depending on type of transmission used and it should be within channel limits of the end point. During encoding, it is important to select the encoding profile Indication Level of Codec (IDC) during video encoding. For H.264 coding there are 17 standard profiles are used for specific applications, of these 17 profiles, baseline profile is used for mobile transmission over network. As the baseline (66 IDC), profile is mainly used for video conferencing, in mobile applications [11]. Baseline profile is mostly used at encoder process for handheld and portable devices. For real time requirement of low-end applications for mobile platform or hand held devices, H.264 under baseline profile coding will attain acceptable video quality [12]. In [13] the author has compared two H.264 coding software’s to identify the performance for different bit rates. For these two popular coding software’s JM reference software and IPP coder are considered. It was stated that JM reference software is good at video quality at less bit rate and the compression size of this coding is acceptable compared with Intel IPP coder. So finally, we have decided to implement our encoding and decoding part of our video generation in JM reference software that satisfies the requirement of our thesis work.
After compression of the video, the different quality measurement metrics subjective and objective are supporting the researcher in many ways. In [14] the quality of video is introduced in two ways, one is during the coding of source video i.e. compression of video before transmission and the other is channel distortion. Transmission of video in bit stream over channel will results in loss of packets or bit error, which results in degradation of the video at the receiver end. The ultimate problem is at the decoding, to decode the video for attaining the properties of original source video either in spatial or/and temporal resolution and it depends on the bit rate that was selected by the network [5]. For this, we have decoded the received video and up scaled to attain the source video properties of temporal and spatial resolutions. So identifying the best up scaling technique for the encoded will solve the problem of degradation of video after transmission. Results of the subjective test will give the conclusion of our thesis work.

According to [18] SAMVIQ was developed for video quality assessment that is most suited for video streaming to PC, mobile and PDA terminals. As the proposed SAMVIQ test methodologies and grading scales were standardized in ITU-R, but still some type of sensitivity and the reliability are not been considered like display type, display distance, hidden reference, hundred point scaling.. Due to this, different organizations are using different scaling grades. SAMVIQ methodology is consuming more assessing time when compared to other methods due to its playback option of sequence in the test. Considering this drawback of more assessing time of SAMVIQ, the developed PVQA tool in this research was made to play the video for one time and has to be graded within the time limit that will be explained in chapter 4. The following chapter 3 and 4 describes the generation of video sequences and the development of PVQA tool respectively.
GENERATION OF VIDEO SEQUENCES
Chapter 3

3. Generation of Video Sequences

3.1. Introduction for Video Generation

This chapter provides the complete information of test video generation for PVQA test. Our method of generating the test sequences are processed in nine steps. For generating these test sequences FFMPEG, Virtual Dub and JM reference software were used.

3.2. Process of Generating Test Video Sequences

This section will give steps for the generation of video sequences and explained in section 3.3

- Converting original raw video size 1028x720 with 50fps YUV 4:2:0 to raw codec AVI with VGA size and 25fps.
- AVI video is down scaled in spatial and temporal resolutions.
- Converting both the scaled and non-scaled AVI files to YUV for encoding.
- Encoding YUV files to H.264 bit stream for channel transmission using baseline profile at 300kbps and 600kbps bit rates respectively.
- Decode H.264 to YUV: called reconstructed YUV.
- Converting reconstructed YUV to raw AVI file.
- Up scaling the raw video in spatial and temporal resolution files to attain the source video properties: VGA size with 25fps.
- Clipping the video length to right length to overcome the initial I frame effect.
- Finally, test video sequences are ready for test.

3.3. Test Video Sequences Process

This section provides detailed description for the generation of test video sequences and workflow. Figure 1 shows the test video samples that are considered for sequence generation: Park run ‘a’, Ducks Take Off ‘b’, Mobile Calendar ‘c’ and Shields ’d’.
3.3.1. Conversion of Raw Video to Obtain Source Video

Initially all the raw YUV format videos are converted to AVI format for down scaling, for this we have used FFMPEG to convert the codec container from YUV to AVI of length 10 s each. For our requirement the spatial resolution of all the raw AVI videos are scaled from 1028x720 to 640x480(VGA) in FFMPEG, and the temporal resolutions 50fps is reduced to 25fps by removing every second frame using Virtual Dub [20]. By doing this we have obtained all the raw AVI video sequences with spatial resolution of VGA and temporal resolution 25fps with 10 sec length. In general for any video sequence the 1st frame is an I frame, an intra coded frame with higher quality, the quality is then reduced gradually during the following the P frames. Due to this the presence of I frame will have impact on subjective test, in order to overcome this we decided not to consider the initial 10 sec of the test video. Therefore, we have augmented the same videos sequence of length 10sec in Virtual Dub to form 20 sec length with same spatial and temporal resolutions. Considering this video sequence as a source video or reference video that is to
be coded and finally before conducting the survey the video length is cropped from 20 sec to 10 sec by removing starting 10 sec, the I frame effect can be solved as explained in section 3.3.6.

3.3.2. Normal Video

For evaluating, we have generated the normal or regular video format without any downscaling of either temporal or spatial resolution respectively. This normal video is a source video encoded under properties of VGA and 25fps at 300kbps and 600kbps in JM software with the parameters shown in table 3 i.e. without any downscale or upscale methods. The generation of these video sequences results in normal transmission of video without any scaling. In this way one set of test video are generated for every test sequence in our subjective test video generation.

3.3.3. Down Scaling Process

All source videos i.e. Ducks take off, Mobcal, Shields and Parkrun with VGA and 25fps with 20sec duration are down scaled by spatial resolution from VGA (640x480) to QVGA (320x240) using FFMPEG using low pass filter, the command for this is given in Appendix B. Similarly, the temporal down scaling is done from 25fps to 12.5fps using Virtual Dub, where the every second frame is removed and the syntax for this frame rate reduction is given in Appendix B.

For encoding the videos in JM software, the input file should be of YUV format. Thus, the down scaled video containers needs to be converted from AVI to YUV.

3.3.4. Coding Process

For encoding the downscaled videos and the not scaled video in JM software, the following parameters were given as inputs to the encoder for obtaining required encoded outputs. The parameters considered for coding the test sequences are seen in table 3.
<table>
<thead>
<tr>
<th>JM Encoder Parameters</th>
<th>Temporal Down Sampled Videos</th>
<th>Spatial Down Sampled Videos</th>
<th>Normal Videos</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input files</td>
<td>.YUV</td>
<td>.YUV</td>
<td>.YUV</td>
</tr>
<tr>
<td>Frames to be encoded</td>
<td>250</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>Frame rate</td>
<td>12.5</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>Source resolution</td>
<td>640x480</td>
<td>320x240</td>
<td>640x480</td>
</tr>
<tr>
<td>Output resolution</td>
<td>640x480</td>
<td>320x240</td>
<td>640x480</td>
</tr>
<tr>
<td>Profile IDC</td>
<td>Baseline=66</td>
<td>Baseline=66</td>
<td>Baseline=66</td>
</tr>
<tr>
<td>Rate control bit rates</td>
<td>300kbps 600kbps</td>
<td>300kbps 600kbps</td>
<td>300kbps 600kbps</td>
</tr>
<tr>
<td>ME search mode</td>
<td>UMHexagon search</td>
<td>UMHexagon search</td>
<td>UMHexagon search</td>
</tr>
<tr>
<td>Output files</td>
<td>H.264 and reconstructed YUV</td>
<td>H.264 and reconstructed YUV</td>
<td>H.264 and reconstructed YUV</td>
</tr>
</tbody>
</table>

Table 3 JM software-coding parameters.

Giving these parameters for JM reference software as input, the generated output files will be of H.264 and reconstructed YUV formats. The output H.264 file is used for mobile transmission since we opted for encoding video under baseline profile. The output file reconstructed YUV is extracted or decoded from H.264 after the transmission over a channel capacity 300kbps and 600kbps respectively. So finally, we have obtained decoded YUV files after transmission. These decoded YUV files or reconstructed YUV files are converted back to AVI for up scaling to obtain the source video properties.

3.3.5. Up Scaling Process

For up scaling, all the reconstructed YUV files will be converted to AVI using FFmpeg, syntax for conversion of container is given in Appendix B. The down scaled (spatial and temporal scaling) videos are processed for up scaling to attain the reference video parameters.

- Spatially down scaled videos are up scaled in Virtual Dub by considering two filter methods:
  - Bicubic Interpolation technique.
  - Nearest Neighbor technique.

Giving the spatially downscaled videos in QVGA as an input to the Virtual Dub and performing the resize filter methods by considering above two techniques for obtaining VGA videos format will complete up-scaling process for spatial resolutions.
Similarly, Temporal down scaled videos are up scaled in Virtual Dub by performing repeat frame method. The temporal down scaled videos with 12.5fps are given as an input for Virtual Dub. These are processed by repeating the successive frame, which will result in output video with 25fps.

Once the up scaling process for spatial and temporal resolution is done with the parameters, the video generation is finished and the final step of solving the I frame effect is to be done.

3.3.6. I Frame Effect

After the up scaling process of the test video sequences, we have clipped the first 10 sec of all generated videos and by using the remaining 10sec (11-20sec) the I frame effect is avoided. By performing this action for all the generated videos, the final test video sequences are ready for the subjective tests.

3.3.7. Test Video Sequences

By performing this process of generation for the test sequences: “Ducks Take Off”, “Mobcal” and “Parkrun” each with four methods (3 scaling methods and 1 normal method) with two different control rates results in twenty-four test sequences. For training purpose “Shields” with four scaling methods and with one control rate are generated. By doing this, finally 28 test video sequences with four reference videos are generated for our project work. An overview of the test sequence generation is shown in figure 2. Figure 3 shows the test video generated for the Parkrun scene.
Figure 2 Overview of the video generation process for subjective tests.
Figure 3: Parkrun sample videos generated for 300kbps and 600kbps.
DESIGN OF PVQA TOOL
Chapter 4

4. Design of the PVQA tool

4.1. Architecture Design of PVQA Tool

The architecture design corresponds with all the elements embedded as a set to perform required process (i.e. SSCQE). The entire software prototype is developed in Java programming language using Eclipse and Net Beans IDE’s (Integrated Development Environments). The software elements included in PVQA tool comprises of database elements, GUI elements, process control elements, VLC player integration files with relevant packages for the entire process of SSCQE setup. In the developed prototype with software elements, supporting Java archive files (.jar), process control codes are aggregated to form a single .jar file (e.g. MOSTOOL.jar). By aggregating to a single file, we can move this .jar file to any system to execute the prototype. The generated .jar file can be embedded to a batch file named ‘.bat’ file for execution in windows environment. The developed prototype can be extended to execute in UNIX environment, depending on the required supported software elements included for successful execution.

4.1.1. JRE System Library

This package named JRE System library is an inbuilt package provided by Eclipse IDE. This package includes:

- resources.jar
- rt.jar
- jsse.jar
- jce.jar
- charsets.jar
- dnsns.jar
- localedata.jar
- sunjce_provider.jar

These jar files consists of classes to provide better programming environment. Some resources like BTH logo or Settings text file can be accessed with classes from resources.jar file. These resources are considered, when ever required.
4.1.2. Referenced Libraries Package

The required support files or jar files are embedded in the referenced libraries package. The included .jar files are as follows:

- jna.jar
- platform.jar
- sqlitejdbc-v056.jar
- vlcj-1.2.0.jar
- jxl.jar

The following figure 4 shows all the .jar files embedded in the development phase.

![Figure 4 jar files embedded for the development of PVQA tool.](image)

4.1.2.1. Java Native Access

The archive file (jna.jar) software contains several packages of classes for interface access for different platforms like Windows, UNIX, Linux and Mac OS.

4.1.2.2. Platform

The platform jar will support the software for different system environments as well as programming environments. E.g., GUI’s are designed in Net Beans IDE and for process logic we use Eclipse IDE.
4.1.2.3. Sqlitejdbc

Sqlitejdbc.jar file is aimed for the database operations like creating a .db (database) file, creating database tables, inserting the data into corresponding tables. The portability and scalability was the main reason to use the Sqlite database.

4.1.2.4. Video Lan Client (VLC)

The VLC is a popular media player suitable for our requirement. In order to embed VLC player, we have to consider vlcj-1.2.0.jar file. This jar file helps to build in VLC player into a GUI. Media player options like play and stop of test sequences can be opted through this jar file for survey requirement.

4.1.2.5. J excel API

J excel API (Jxl.jar) is used to create a function for excel sheet generation with required template. All the Survey results can be represented in excel sheets using this jar file.

4.2. Organization of Architecture Design

The structure of the architecture arrangement of PVQA tool is outlined in figure 5.
Figure 5 Outline representing architecture arrangement of PVQA tool.

All the GUI’s, JAVA classes and JAVA methods needed for the process are defined and declared for our requirement in the concerned packages. Main.java is the main class where our entire process starts. It is located in main package. The nomenclature of the packages is given according to the event driven actions in corresponding packages.

4.3. Database Implementation

The process model of the PVQA tool can be realized with declaration and implementation of database required for the process. Database element utilized in
developing the tool is Sqlite. Features like portability, scalability and simplicity nature to implement RDBMS (Relational Database Management System) influenced to use Sqlite database element tool. The java archive file sqlitejdbc-v056.jar will enable us to implement database file. We name the .db (database file) as mosstool.db file. Whenever the tool is initiated, the code will check for availability of mosdatabase.db file. If the file is available it proceeds to connect the db file using the jdbc connector. If the code does not find mosdatabase.db file, the whole Sqlite jar file proceeds to setup a database file. Figure 6 represents the location of JDBC driver that is required in connecting the .db files.

Figure 6 Location of JDBC driver required for Database operation.
4.3.1. Design of Database Tables

The design of database is depicted in figure 7.

![Database Design Diagram](image)

**Figure 7 Database Design in Mostool.**

The database file in mosdatabase.db consists of nine tables, designed according to the tool requirement. The list of designed database tables is as follows:

- ADMIN TABLE
- USER TABLE
- VIDEO TABLE
- VIDEO GRADING TABLE
- CUSTOM SCALE TABLE
- BINARY SCALE TABLE
- GLOBAL PROPERTIES TABLE
- SURVEY TABLE
- TIMER SETTINGS TABLE

The names of the tables are given corresponding to the data to be stored in them while the tool is running. Sqlite jdbc driver aids to create the respective tables in the process. During the processing session of the tool, these tables are created, retrieved, deleted and updated. These methods help
us to get the tools working in the background. Tables are created for data to be stored in the respective table, e.g. username, age, gender and skill must be saved in USER TABLE. The data can then be retrieved from the table and employed to further processing like adding survey results to particular user from the VIDEO GRADING TABLE. The tables can also be updated or deleted.

The representation of each database table is as follows:

- **Admin Table**

<table>
<thead>
<tr>
<th>ADMIN TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>USER NAME</td>
</tr>
<tr>
<td>PASSWORD</td>
</tr>
</tbody>
</table>

  Table 4 Admin table.

  - ID is the primary key of the table.
  - USERNAME is name of the admin to be entered.
  - PASSWORD is the key to authenticate for admin to give complete privileges to use the tool.

- **User Table**

<table>
<thead>
<tr>
<th>USER TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>NAME</td>
</tr>
<tr>
<td>AGE</td>
</tr>
<tr>
<td>GENDER</td>
</tr>
<tr>
<td>SKILL</td>
</tr>
</tbody>
</table>

  Table 5 User table.

  - ID is auto increment of primary key in integer type.
  - NAME, GENDER and SKILL are of varchar type, determined to be not null.
  - AGE is an integer type set as not null.

- **Video Table**

<table>
<thead>
<tr>
<th>VIDEO TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>FILE NAME</td>
</tr>
<tr>
<td>URL</td>
</tr>
<tr>
<td>FK_SURVEY_ID</td>
</tr>
<tr>
<td>CONSTRAINT</td>
</tr>
</tbody>
</table>

  Table 6 Video table.
- ID is the primary key in integer type.
- Name of the test video sequence will be saved in FILENAME column.
- URL entailed to save the location of the video file.
- FILENAME and URL are of varchar type, determined as not null.
- FK_SURVEY_ID is a foreign key attribute from SURVEY table ID.
- CONSTRAINT of this table is both URL and FK_SURVEY_ID must be unique.

**Video Grading Table**

<table>
<thead>
<tr>
<th>VIDEO GRADING TABLE</th>
<th>ID INTEGER PRIMARY KEY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FK_USER_ID INTEGER NOT NULL (reference to USER TABLE ID)</td>
<td></td>
</tr>
<tr>
<td>FK_VIDEO_ID INTEGER NOT NULL (reference to VIDEO TABLE ID)</td>
<td></td>
</tr>
<tr>
<td>FK_SURVEY_ID INTEGER NOT NULL (reference to SURVEY TABLE ID)</td>
<td></td>
</tr>
<tr>
<td>GRADE INTEGER NOT NULL</td>
<td></td>
</tr>
<tr>
<td>CONSTRAINT UNIQUE FK_USER_ID, FK_VIDEO_ID, FK_SURVEY_ID</td>
<td></td>
</tr>
</tbody>
</table>

Table 7 Video Grading table.

- ID is the primary key of this table.
- FK_USER_ID, FK_VIDEO_ID and FK_SURVEY_ID are the foreign keys corresponding to ID’s of USER TABLE, VIDEO TABLE and SURVEY TABLE.
- GRADE is an integer type, where grades are stored in this attribute of table.
- The CONSTRAINT of this table is FK_USER_ID, FK_VIDEO_ID and FK_SURVEY_ID and it must be unique.

**Custom Scale Table**

<table>
<thead>
<tr>
<th>CUSTOM SCALE TABLE</th>
<th>ID INTEGER PRIMARY KEY</th>
</tr>
</thead>
<tbody>
<tr>
<td>FK_SURVEY_ID INTEGER NOT NULL (reference to SURVEY TABLE ID)</td>
<td></td>
</tr>
<tr>
<td>MINSCALE VARCHAR NOT NULL</td>
<td></td>
</tr>
<tr>
<td>MIDSSCALE VARCHAR NOT NULL</td>
<td></td>
</tr>
<tr>
<td>MAXSCALE VARCHAR NOT NULL</td>
<td></td>
</tr>
</tbody>
</table>

Table 8 Custom Scale table.
- ID is an integer type and primary key of the CUSTOM TABLE.
- FK_SURVEY_ID is an integer type and foreign key to ID of SURVEY TABLE.
- MINSCALE, MAXSCALE and MIDSACLE are the three attributes of varchar type.

➢ Binary Scale Table

<table>
<thead>
<tr>
<th>BINARY SCALE TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>FK_SURVEY_ID</td>
</tr>
<tr>
<td>POSITIVE_SCALE</td>
</tr>
<tr>
<td>NEGATIVE_SCALE</td>
</tr>
</tbody>
</table>

Table 9 Binary Scale table.

- ID is an integer type and primary key of the BINARY_TABLE.
- FK_SURVEY_ID is an integer type and foreign key to ID of SURVEY TABLE.
- POSITIVESCALE and NEGATIVESCALE are two attributes of varchar type.

➢ Global Properties Table

<table>
<thead>
<tr>
<th>GLOBAL PROPERTIES TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
</tr>
<tr>
<td>KEY</td>
</tr>
<tr>
<td>VALUE</td>
</tr>
</tbody>
</table>

Table 10 Global Properties table.

- The primary key of this table is ID.
- KEY and VALUE is a pair of VARCHAR type to collect data correspondent to the request e.g. If KEY is CURRENT_SURVEY_NAME, then VALUE will be the name of the survey given by the admin.
- KEY should be unique.
➤ **Survey Table**

<table>
<thead>
<tr>
<th>SURVEY TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID INTEGER PRIMARY KEY</td>
</tr>
<tr>
<td>SURVEYNAME VARCHAR NOTNULL (unique)</td>
</tr>
<tr>
<td>SCALETYPE VARCHAR NOTNULL</td>
</tr>
<tr>
<td>FK_TIMER_ID INTEGER NOTNULL(unique)</td>
</tr>
<tr>
<td>VIDEOLOCATION VARCHAR NOTNULL</td>
</tr>
</tbody>
</table>

Table 11 Survey table.

- ID is the primary key.
- SURVEYNAME, SCALETYPE and VIDEO LOCATION are type VARCHAR.
- Name of the survey is stored in the attribute of SURVEYNAME, so it is determined as unique.
- The ID of TIMER_SETTINGS_TABLE mentioned in the bellow will be addressed to have a foreign key in this table i.e. FK_TIMER_ID. It should be defined as unique because all the videos of a survey should follow single timer settings.

➤ **Timer Settings Table**

<table>
<thead>
<tr>
<th>TIMER SETTINGS TABLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID INTEGER PRIMARY KEY</td>
</tr>
<tr>
<td>TIMER TYPE VARCHAR NOTNULL</td>
</tr>
<tr>
<td>T1 INTEGER NOTNULL</td>
</tr>
<tr>
<td>T2 INTEGER NOTNULL</td>
</tr>
<tr>
<td>T3 INTEGER NOTNULL</td>
</tr>
</tbody>
</table>

Table 12 Timer Settings table.

- ID is the primary key.
- Type of timer which admin intend to use is saved in TIMERTYPE attribute.
- T1, T2 and T3 are the three attributes of integer type, which saves the timer values for corresponding survey settings given by admin. T1 cannot be notified as NOTNULL.
4.4. GUI Implementation

The Graphic User Interface of our application was implemented using NetBeans Integrated Development Environment (IDE). All the classes representing the GUI’s are defined in two packages namely (1) ui.admin and (2) ui.user. According to the PVQA process model there are two modes for the SSCQE experimental setup i.e. admin mode and subject mode. Therefore, all GUI’s for admin mode are defined in ui.admin package and the entire user mode GUI’s are defined in ui.user package. When the application is initiated with creation of database in Sqlite with mosdatabase.db file, a GUI interface is established for the user to interact with the tool. The first interface, thus created initially is coded with the name MainScreen.java.

4.4.1. Main Screen.java

The GUI representation of the main screen is represented in figure 8. All the event driven programming parts are coded in this class.
This main screen has two radio buttons and three click buttons. The two radio buttons are created for Survey conductor to select any one of the modes. The INSTRUCTIONS Button is used to generate a text file with instructions on how to use the PVQA tool. All the instructions like time settings, time-elapse settings, etc., for admin as well as subject mode are provided in that text file. This text file works as a user guide to the PVQA tool. The EXIT Button is used to exit the operation and OK Button to continue for the respective radio button selection. This MainScreen.java file is embedded in .main package. All the required events are coded in MainScreen.java.

4.4.2. AdminLoginScreen.java

After selecting admin radio button and pressing OK button, a new GUI is generated for further action. The newly generated GUI is a login screen for authentication purposes. The GUI consists of certain text labels, one text field, one password field and two click buttons. The GUI representation of AdminLoginScreen.java is shown in figure 9.

![Figure 9 Admin Login Screen, generated after selecting admin mode.](image)

The Admin login screen will have an OK and a CANCEL click Buttons. The OK button is intended to start the authentication process for admin mode. CANCEL button is exit the present GUI and return the Main Screen GUI.
4.4.3. Admin Settings form and New Survey Panel

After the authentication of admin mode, a new GUI is generated which is a combination of classes AdminSettingsForm.java and NewSurveyPanel.java. It is the main GUI for the admin to set up survey settings according to his/her requirement. This GUI consists of four phases to provide Survey admin with options to conduct the survey. Each phase consists of respective text fields, radio buttons and click buttons. Help menu is attached to the give the information regarding PVQA Tool development team. The representation of Survey Settings GUI is shown in figure 10. The four phases implemented in this GUI are as follows:

- Name of the survey.
- Scale settings.
- Time elapse settings.
- Video location.
4.4.3.1. Name of the survey phase notifies admin to give a name to the survey in the text field of GUI. Name of the survey is saved in the corresponding database table and utilized for the generation of test results in excels pages.

4.4.3.2. Scale settings phase consists of certain radio buttons which are grouped, intended to select only one scale for each survey. There are four types of scales available in this phase listed in table 14. Continuous Scale is for using a slider as tool to grade the survey; it consists of 0 to 100 points for grading. MOS five-point scale is an ITU-T recommended scale. There are
two types, namely Impairment scale and quality scale in MOS scale shown in table 13. Binary scale is a two level grading scale designate for subject to grade as binary level. Binary Scale consists of two radio buttons and two text fields, so that those radio buttons act as selecting options when subjects are grading. Those text fields are enabled for the admin to edit text, in order to show the text to subjects while grading e.g. Acceptable and Un-Acceptable. Custom Scale is similar to the binary scale, but having an extra radio button with text filed. Custom Scale has three levels for grading with three text fields for subjects to grade e.g. BEST, GOOD and BAD. These four types of scales are used as options for each survey and the admin needs to choose one.

<table>
<thead>
<tr>
<th>Five-Grade Scale</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Quality</td>
<td>Impairment</td>
</tr>
<tr>
<td>5 Excellent</td>
<td>5 Imperceptible</td>
</tr>
<tr>
<td>4 Good</td>
<td>4 Perceptible, but not annoying</td>
</tr>
<tr>
<td>3 Fair</td>
<td>3 Slightly annoying</td>
</tr>
<tr>
<td>2 Poor</td>
<td>2 Annoying</td>
</tr>
<tr>
<td>1 Bad</td>
<td>1 Very Annoying</td>
</tr>
</tbody>
</table>

Table 13 ITU-R BT.500-12 Five Point Scaling.

<table>
<thead>
<tr>
<th>SCALE NAME</th>
<th>GRADE LEVELS</th>
<th>GRADING OPTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hundred Point Scale</td>
<td>100</td>
<td>Grade Pointer</td>
</tr>
<tr>
<td>Five Pointer Scale (Quality)</td>
<td>5</td>
<td>Radio button</td>
</tr>
<tr>
<td>Five Point Scale (Impairment)</td>
<td>5</td>
<td>Radio button</td>
</tr>
<tr>
<td>Binary Scale</td>
<td>2</td>
<td>Radio button with text</td>
</tr>
<tr>
<td>Custom Scale</td>
<td>3</td>
<td>Radio button with text</td>
</tr>
</tbody>
</table>

Table 14 Types of scales available for subjective grading.

4.4.3.3. Time elapse settings phase is designed for admin to select playing duration of the test sequences, grading time and pause time. In this phase, three modes of time elapse settings are built for survey settings sake. The three modes of elapse settings are ITU mode, Custom mode and Full-length video mode. All modes have an elapse setting with three built-in timer settings T1, T2 and T3, except for Full-length video mode. T1 is the duration of each test sequence to be played. T2 is the time duration where subjects are asked to grade the assessed video and T3 is the pause time for subject before next sequence. ITU mode has fixed timer values as T1=10, T2=15 and T3=5. Whereas, in Custom mode all the three timer values can be customized for desired values. Custom mode helps to conduct variable number of surveys with different timer values. In Full-length video mode, the video is played for its total duration, whereas T2 and T3 can be customized for required values.
The description of three timer settings can be analyzed in Table 15.

<table>
<thead>
<tr>
<th>NAME</th>
<th>T1</th>
<th>T2</th>
<th>T3</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITU-R BT.500-12</td>
<td>30s (Static)</td>
<td>10s (Static)</td>
<td>5s (Static)</td>
</tr>
<tr>
<td>Custom</td>
<td>VARIABLE</td>
<td>VARIABLE</td>
<td>VARIABLE</td>
</tr>
<tr>
<td>Full length video</td>
<td>VARIABLE</td>
<td>VARIABLE</td>
<td>VARIABLE</td>
</tr>
</tbody>
</table>

Table 15 Timer settings for PVQA test.

4.4.3.4. In video location phase, the text field is to show the location of the test sequences folder. It also contains click button to add the location of the test sequences folder.

There are OK, RESET and START SURVEY click buttons in this GUI. RESET button is intended to reset all the survey settings, when admin wants to change entirely. START SURVEY click button is disabled initially because it can be used only after the settings are saved. All the settings are saved with OK button event. Therefore, after clicking the OK button all the settings are saved and altered with a pop-up message. Then after acknowledging with the alert message START SURVEY button is enabled. By selecting START SURVEY button, admin can access main Screen GUI to select the Subject mode. The acknowledgment of saved Survey settings is shown in figure 11.
After selecting the Start Survey Click button, the MainScreen.java GUI is accessed again. Then from the main screen GUI it will be possible to access subject mode.

4.5. Subject Details Screen

Subject details screen is entailed to give details of the subject. The GUI of Subject details screen is designed in the java GUI class Userdetails.java. This subject details screen consists of three stages “I Agree” radio button, OK and Start Survey click buttons. This GUI consists of three phases, which subjects need to give the required details to. The first phase is to accept a policy about subject answers are intended for research purpose. By clicking radio button, the subject will enable the OK click button in the third phase. In the second stage, subjects have to give details like age, gender and skill level. After filling these details he may proceed to the third phase. The third phase has three click buttons, i.e. BACK, OK and Start Survey. BACK button is to abort the operation. OK button is to save the subjects information in User database table and enable the Start Survey button. After the pressing OK button, the subject is enabled to enter third phase. The third phase is precisely with a
start survey click button. All the test sequence with the concerned timer settings is started by clicking Start survey click button. All the event driven movements according to logics are coded in Userdetails.java class. The following figure 12 and figure 13 shows the depiction of Subject GUI of Subject details.

Figure 12 Subject details screen before accepting security policy button.
Subjects are employed to use mouse for grading session. During T1 duration test sequence is played with LAB environment conditions. On T2 time duration test sequence is meant to be graded the subject i.e. getting the subject is giving a score corresponding to its perceived quality. The following figure 14 shows an example of the screen during T1 session.
The figure 14 shows the reference video sequence to express the reference sequence to the subject. All the test sequences whether reference or non-reference sequences are played T1 duration. In our experimental setup, we fixed the resolution of test sequences to VGA format, i.e. 640x480. Our PVQA tool can be modified in NetBeans IDE according to the resolution size like CIF, QCIF, QVGA, etc depending on survey requirement. During the T1 session, the grade panel is disabled. According to the main requirement subjects are not empower to grade while video is playing. The Non-reference video is shown in the figure 15. Subject has to analyze the quality of the Non-reference video psychologically and give his/her perception on it. Differences of reference video and Non-reference video can be examined between figure 14 and 15.
The reference text is not seen in this Non-reference sequence. The duration of T2 session, where subject is employed to grade the video is shown in the figure 16. Though there are four scales inbuilt in our tool, the continuous scale where resolved for our survey, as it can provide accurate results. Statistical analysis and graphical representation can be achieved effectively with using continuous scale. The score range of continuous scale is 0-100. If subject does not score any of sequence, the default value to be assigned is “-1”. Five point MOS (Mean opinion score) are assumed as levels such as, excellent, good, fair, poor and bad. The continuous scale also has the Score panel, so that subject has a quality scale reference.
After the T2 session, T3 session will be initiated for the reloading of the next video in line. We coded the tool to shuffle the playlist, as an optional requirement for remaining surveys. During T3 session subject can take relief for a few seconds, so that they can concentrate the next coming video. After the grading of all videos are done a pop up message is displayed to the subject so that he/she knows that the survey session is terminated. The message or alert after completing each subject will be “thank you for your time”. The following GUI figure 17 presents the termination stage of each subject.
4.6. Architecture Representation

Our PVQA tool can be expressed in three models to give an overview of the whole process i.e.

- Design Model
- Use Case Model
- Process Model

4.6.1. Design Model

The PVQA tool was designed based on EJB (Enterprise Java beans) model, but limited to the standalone process. The design model of PVQA denotes five divisions namely database, entity, data access objects, process logic and interfaces. Database division denotes the entire database tables mentioned in the database implementation where all the data required is stored while tool is running. Entity division helps us to denote setters and getters methods for the required operation. Data access objects (DAO) is the section where all the object persistence, data accessing logic and entity creation methods are declared and
defined. This is the head role of database retrieving and updating. Logic section has three types of logics namely GUI logic, Event logic and Process logic. This is the heart of the whole model. All methods described in the whole process are utilized in a logical way to reach the substantial process. Admin and subject are two interfaces of our tool. Admin interface with process logic are separately designed from the user interface. The event flow logic of both interfaces is projected in Event logic. The design model of PVQA tool is seen in figure 18.

![Figure 18 Design model of PVQA tool.](image)

4.6.2. Use Case Model

As there are two interfacing modes in PVQA tool, i.e. ADMIN and SUBJECT, there are two use case scenarios in the process mode.

- Admin mode.
- Subject mode.
4.6.2.1. Admin Mode

Figure 19 gives the overview of Admin mode use case interface.

Figure 19 Use case in Admin mode.
4.6.2.2. Subject Mode

Use case interface in subject mode is seen in figure 20.

Figure 20 Use case in Subject mode.
Process Model

The process of the PVQA tool can be divided into three divisions named as:

- Initialization mode.
- Admin mode.
- Subject mode.

4.6.3.1. Initialization Mode

The initialization mode contains the sequence of processes run when the tool is initiated. When the tool is initiated with the jar file, the process starts with checking for a mostool.db file in DBconstants.java class. Then if the db file exists, it will keep all the tables in the db file and ready to update with a new data into the tables. If the db file does not exist, it will construct new db file with relevant tables mentioned above in database implementation using DBconstants.java file. A flow chart of initialization mode in PVQA tool is seen in figure 21. Parallelograms described in the flow chart deals with input and output operation of GUI’s. Rectangles are execution process steps and Rhombuses are condition verification steps.
Figure 21 Flow chart of initial mode in PVQA tool.
4.6.2.3. Admin Mode
Admin mode deals with the process of admin settings, the flow chart represents the sequence of steps in admin mode shown in figure 22.

Figure 22 Flow chart of Admin mode process in PVQA tool
Save survey settings using saveToDatabase method.
Create survey entity
Set survey name using setSurveyname().
Set survey Scale type using setSurveyscale type().
Set timer Settings using getTimer().
Save details using
Madminsettings.Savesettings (survey, timer, Binary scale and Custom scale)

STEP 1(saveSettings method include)
Validate if video with name “original” exists are not

STEP 2(saveSettings method include)
Save the video files in an ArrayList name VideoFile.

STEP 3(saveSettings method include)
Assigning global properties (i.e. key and value) For Entity

Admin settings form.java + New Survey panel / ExistingSurvey Screen java
start survey button enabled

Press Start Survey

MainScreen.java
STEP 1 Verifying for Reference videos

Validate ifOriginalVideoExists method (file videoFile)

Create array list with name "dirList"

IF videoFile is directory

YES

Create listfiles array[]
Listfiles = VideoFile.list files
Video is the variable in Listfiles

Video <= List files

NO

stop

YES

Check video is directory

NO

Exception handling

YES

Add video to dirList i.e. dirList.Add(video)

Video++
STEP2: Saving Video files

video File is variable of array listfiles.

Videoile < = Listfiles

NO

Exception handling

YES

Use traverse method (video file, survey)

Videoile ++

Check Video is directory

YES

Save videos

NO
STEP 3: Assigning key value pair of Global properties

1. Find global properties (entity) by FindByKey method.
2. Check global properties are NULL
   - NO: Exception handling with message “could not set survey as current survey”
   - YES: Globalproperty.deleteentityByID
3. Create New Entity globalproperties with methods
       Globalproperty.setkey()
       Globalproperty SetValue()
       end Globalproperty.createEntity()
Subjective mode deals with the PVQA tool process for subject performance, the flow chart in figure 23 represents the sequence steps in subject mode.

Figure 23 Flow chart of Subjective mode process in PVQA tool.
UserScreen.java (includes)
init.component()
Get current survey details()
Load scales and questions
Set timer:
Timer = current survey details.timer()
    T1 = timer.getT1()
    T2 = timer.getT2()
    T3 = timer.getT3()
PlaylistCounter (list of video files)
    T2 timer = new timer (T2 * 1000, new T2 timer)
    T3 play timer = new timer (T3 * 1000, new T3 timer)
    TempT2 = T2
    TempT3 = T3 - 1
    Grade panel set visible (false)
    Reference label set visible (true)

StartSurvey() method
implments
T3 PlayTimer.start()
T3 Label Timer.start()
System.out.print currently playing video + current video.getTitle();

Assign file name and extension as strings.

If file name is "original"
  Set reference label (true)
  Set grade panel (false)

Remove the previous video played in media list (i.e. VLC)

If \( t > 0 \)
  Insert current video using mediaList.insertMedia() method with attributes [0, currentVideo.getUrl() + t2]

Else
  Insert current video using mediaList.insertMedia() method with attributes [0, currentVideo.getUrl()]

If Video is first
  \( t \)

Else
  \( t \)
6

T2_TimerStop();
T2Label_timer_stop();
Set_label
(-textT2=“sec”)

If user select
the grade
value

NO
Set Value as “-1”

YES
Grade video

Display survey panel

Set if First Video = false;
Temp T3=3-1;
Teplaytimer=Restart[];
T3label_timer=Restart[].

7

If current
video exist

Assign video file and video extension into string

If file name is original

NO

YES
Enable grade panel
Enable text label

Enable grade Panel
Disable text panel
EXPERIMENTAL SETUP
Chapter 5

5. Experimental Setup

5.1. Outline

This chapter will provide the information on experimental setup conducted for test video sequences using PVQA tool, and identify how this tool helps in conducting the subjective tests.

5.2. Test Environment

For conducting the subjective tests, we used Blekinge Institute of Technology, Radiocom laboratory under laboratory environment conditions recommended by ITU-R BT.500-12 [1]. For this subjects test, two Desktop computers with the configuration given below in figure 24 are used. Readings for laboratory environment conditions mentioned in table 17 with room temperature and complete survey took two working days for 39 subjects. This lab has controlled lighting conditions that makes ease for maintaining the required environment. The background of the desktop is with light color, which will not distract the assessors during the test.

![Figure 24 System configurations for PVQA test.](image)

Our method of conducting the survey is by considering the methodology of ITU-R SSCQE and some recommendations of SAMVIQ [17]. Table 16 provides the recommendations that are been considered for our subjective tests.
<table>
<thead>
<tr>
<th>Parameters</th>
<th>SSCQE</th>
<th>SAMVIQ</th>
<th>PVQA Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Explicit reference</td>
<td>NO</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Hidden reference</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Scale</td>
<td>Bad to Excellent (1 to 5 point)</td>
<td>Bad to Excellent (0 to 100 point)</td>
<td>Bad to Excellent (0 to 100 point)</td>
</tr>
<tr>
<td>Sequence length</td>
<td>10s</td>
<td>10s</td>
<td>10s</td>
</tr>
<tr>
<td>Picture format</td>
<td>All</td>
<td>All</td>
<td>All</td>
</tr>
<tr>
<td>Presentation of test material</td>
<td>Once</td>
<td>Several concurrent</td>
<td>Once</td>
</tr>
<tr>
<td>Voting</td>
<td>Test sequences</td>
<td>Test sequences and reference</td>
<td>Test sequences and hidden reference</td>
</tr>
<tr>
<td>Possibility to change the vote before proceeding</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Continuous quality evaluation</td>
<td>Yes (moving slider in a continuous way)</td>
<td>No</td>
<td>Yes (moving slider in a continuous way)</td>
</tr>
<tr>
<td>Assessor per display</td>
<td>One or more</td>
<td>One</td>
<td>One</td>
</tr>
<tr>
<td>Display</td>
<td>Mainly TV</td>
<td>Mainly PC</td>
<td>Mainly PC</td>
</tr>
</tbody>
</table>

Table 16 List of parameters considered from ITU-R and SAMVIQ recommendations.

<table>
<thead>
<tr>
<th>Measurements</th>
<th>System1</th>
<th>System2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Day1</td>
<td>Day2</td>
</tr>
<tr>
<td>Luminance of display when showing complete white</td>
<td>15.0 cd/m²</td>
<td>15.5 cd/m²</td>
</tr>
<tr>
<td>Luminance of display when showing complete black</td>
<td>1.0 cd/m²</td>
<td>1.0 cd/m²</td>
</tr>
<tr>
<td>Luminance of room light falling on face</td>
<td>27.6 lux</td>
<td>27.2 lux</td>
</tr>
<tr>
<td>Luminance of display light falling on face</td>
<td>18 lux</td>
<td>18.6 lux</td>
</tr>
</tbody>
</table>

Table 17 Experimental setup measurements.

### 5.3. Test Procedure

When all parameters and lab environments recommended by ITU-R BT.500-12 are satisfied, the survey has been conducted. Initially, subjects are guided with the operation of the tool. The assessor is given information regarding the video sequence play order i.e. number of video sequences are shown and the subject is expected to grade the visual quality of each of these in the range of 0 to 100. A reference video is initially shown, which has the best possible quality and cannot be graded. Once the reference video shown, the assessor is able to see test video sequences that shown only once and he/she is expected to grade each of them within 10s after the last frame.
is shown. Once the grade time is finished, there will be a pause time of 5 sec before the next sequence is played. Like this for every scene, a reference video played initially, followed by test videos. A total of 42 video sequences were shown to each subject that includes references, test sequences and hidden references. Figure 25 shows the instructions given to the subjects during survey.

Figure 25 PVQA survey instructions giving to the subjects.

As a training session, the assessor is shown the shields video sequences first and these grades will not be considered for conclusion. For this, the tool is developed in such a way that training sequences will be played initially and then the remaining sequences will be played. For every scene, the test video sequence consists of Source/original video, generated video formats and a hidden reference. The presence of training videos or hidden references is not stated to the assessor and it is not considered for the final survey results. The presence of hidden reference will improve the effectiveness of grading results as stated in [17]. For every scene (Mobcal/Parkrun/Ducks Take Off/ Shields), the generated video formats will be randomized from subject to subject. Like this, we have conducted survey of 39 subjects in the campus with different age groups, skill levels, country of origin and gender. Finally, the results generated by the PVQA tool will be sorted in chapter 6. The overall video sequence flow is shown in the following figure 26.
Figure 26 Video sequences flow in PVQA tool.

T1 = 10 sec (Scene length)
T2 = 10 sec (Grading time)
T3 = 5 sec (Pause time)
N = number of video scenes
n = number of formalized sequences
RESULTS
&
CONCLUSION
Chapter 6

6. Results

6.1. Outline

This chapter provides the results generated from the PVQA tool in the performed subjective tests and its statistical analysis.

6.2. Subjective Test Outputs

After conducting the subjective test for the generated video formats, finally the excel sheet generated by the PVQA tool is considered. The tool is designed in such a way that the grading values of the videos per each subjective test will be updated in excel sheets when the current session is finished. These excel sheets gives complete track of the conducted subjective tests. There is a possibility of outliers in the subjective test which could affect the results. Outliers are those who not assess or assess with wrong perception etc [21]. After conducting the survey tests for 39 assessors, we have removed the subjects who not assessed all videos, i.e. having the score ‘-1’ for one or more of the videos, and considered these as outliers. In the survey results, 7 subjects were removed and not considered for analysis. We have made statistical analysis in order to give the conclusions for our proposed video format transmission.

First, the average of the grades for each video sequence was calculated together with the standard deviation and the 95% confidence interval is shown in table 18.

The acronym used for the type of video formats that are considered for the survey results is given by the syntax X_UV_YZ_ABC.

‘X’ refers to the type of video scenes that are used.

- Mobcal (M)
- Parkrun (P)
- Ducks Take Off (D)

‘UV’ refers to the type of scaling technique used.

- Spatial Scaled (SS)
- Temporal Scaled (TS)
- Normal Technique (NOR)
‘YZ’ denotes the method used for specific up-scaling technique.

- Bicubic Interpolation (BC)
- Nearest Neighbor (NN)
- Repeat Frame (RF)

‘ABC’ denotes the bit rate used for the encoding.

- Bit rate 300kbps (300)
- Bit rate 600kbps (600)

For e.g. D_SS_BC_600: Ducks Take Off formatted based on SS technique using Bicubic Interpolation method for 600kbps bit rate. Similarly M_NOR_300: Mobcal video formatted as a normal video without any scaling for 300kbps bit rate.
<table>
<thead>
<tr>
<th>Video Formats</th>
<th>Average Score</th>
<th>Standard Deviation</th>
<th>Confidence Interval 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>D_Ts_RF_300</td>
<td>27.5</td>
<td>14.8</td>
<td>5.1</td>
</tr>
<tr>
<td>D_Ts_RF_600</td>
<td>33.1</td>
<td>16.4</td>
<td>5.7</td>
</tr>
<tr>
<td>D_SS_BC_300</td>
<td>29.8</td>
<td>19.1</td>
<td>6.6</td>
</tr>
<tr>
<td>D_SS_BC_600</td>
<td>37.9</td>
<td>15.6</td>
<td>5.4</td>
</tr>
<tr>
<td>D_SS_NN_300</td>
<td>25.0</td>
<td>17.0</td>
<td>5.9</td>
</tr>
<tr>
<td>D_SS_NN_600</td>
<td>36.7</td>
<td>15.8</td>
<td>5.5</td>
</tr>
<tr>
<td>D_NOR_300</td>
<td>30.6</td>
<td>14.7</td>
<td>5.1</td>
</tr>
<tr>
<td>D_NOR_600</td>
<td>33.1</td>
<td>15.9</td>
<td>5.5</td>
</tr>
<tr>
<td>D_REF_300</td>
<td>81.2</td>
<td>13.6</td>
<td>4.7</td>
</tr>
<tr>
<td>D_REF_600</td>
<td>87.3</td>
<td>9.6</td>
<td>3.3</td>
</tr>
<tr>
<td>M_Ts_RF_300</td>
<td>54.4</td>
<td>21.7</td>
<td>7.5</td>
</tr>
<tr>
<td>M_Ts_RF_600</td>
<td>59.1</td>
<td>19.1</td>
<td>6.6</td>
</tr>
<tr>
<td>M_SS_BC_300</td>
<td>41.4</td>
<td>14.7</td>
<td>5.1</td>
</tr>
<tr>
<td>M_SS_BC_600</td>
<td>52.0</td>
<td>16.9</td>
<td>5.9</td>
</tr>
<tr>
<td>M_SS_NN_300</td>
<td>40.0</td>
<td>15.5</td>
<td>5.4</td>
</tr>
<tr>
<td>M_SS_NN_600</td>
<td>46.2</td>
<td>17.8</td>
<td>6.2</td>
</tr>
<tr>
<td>M_NOR_300</td>
<td>61.9</td>
<td>20.2</td>
<td>7.0</td>
</tr>
<tr>
<td>M_NOR_600</td>
<td>70.5</td>
<td>17.0</td>
<td>5.9</td>
</tr>
<tr>
<td>M_REF_300</td>
<td>79.9</td>
<td>16.5</td>
<td>5.7</td>
</tr>
<tr>
<td>M_REF_600</td>
<td>80.4</td>
<td>13.3</td>
<td>4.6</td>
</tr>
<tr>
<td>P_Ts_RF_300</td>
<td>27.2</td>
<td>17.2</td>
<td>6.0</td>
</tr>
<tr>
<td>P_Ts_RF_600</td>
<td>28.0</td>
<td>16.7</td>
<td>5.8</td>
</tr>
<tr>
<td>P_SS_BC_300</td>
<td>16.2</td>
<td>10.8</td>
<td>3.8</td>
</tr>
<tr>
<td>P_SS_BC_600</td>
<td>35.1</td>
<td>15.0</td>
<td>5.2</td>
</tr>
<tr>
<td>P_SS_NN_300</td>
<td>16.2</td>
<td>11.4</td>
<td>3.9</td>
</tr>
<tr>
<td>P_SS_NN_600</td>
<td>33.0</td>
<td>14.5</td>
<td>5.0</td>
</tr>
<tr>
<td>P_NOR_300</td>
<td>24.1</td>
<td>13.6</td>
<td>4.7</td>
</tr>
<tr>
<td>P_NOR_600</td>
<td>25.9</td>
<td>14.3</td>
<td>4.9</td>
</tr>
<tr>
<td>P_REF_300</td>
<td>85.4</td>
<td>11.1</td>
<td>3.8</td>
</tr>
<tr>
<td>P_REF_600</td>
<td>83.1</td>
<td>12.5</td>
<td>4.3</td>
</tr>
</tbody>
</table>

Table 18 Statistics of per video with average score and Confidence interval 95%

The graphical representation of the averages and the 95% confidence interval of scores for the video sequences are shown in figure 27 and figure 28, for 300kbps and 600kbps.
Figure 27 Average scores and 95% CI for video sequences generated for 300kbps

Figure 28 Average scores and 95% CI for video sequences generated for 600kbps.
One way of evaluating the proposed video format techniques (i.e. unrelated to the video content) is done by calculating the average and the CI of 95% for all the formatted techniques. For each technique, we have 3 video scenes with 32 subjective grindings \((1 \times 3 \times 32 = 96\) grades\) for each bit rate. The calculated values are shown in table 19.

<table>
<thead>
<tr>
<th>Format techniques</th>
<th>Average</th>
<th>Standard deviation</th>
<th>CI 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>300kbps</td>
<td>600kbps</td>
<td>300kbps</td>
</tr>
<tr>
<td>SS-Bicubic Interpolation</td>
<td>29.1</td>
<td>41.7</td>
<td>18.3</td>
</tr>
<tr>
<td>SS-Nearest Neighbor</td>
<td>27.1</td>
<td>38.6</td>
<td>17.7</td>
</tr>
<tr>
<td>TS-Repeat Frame</td>
<td>36.4</td>
<td>40.1</td>
<td>22.3</td>
</tr>
<tr>
<td>Normal Technique</td>
<td><strong>38.9</strong></td>
<td><strong>43.2</strong></td>
<td>23.2</td>
</tr>
</tbody>
</table>

Table 19 Statistics of proposed formatting technique with confidence interval 95%.

According to subjects grading with respect to scaling techniques, we found that most of the subject’s preferred normal video format rather than any of the three scaling techniques.

The above two methods of evaluation per video and per formatting technique gave some basis for analysis. To identify the better technique based on video content; we have calculated the average and the CI of 95% for each formatting technique for each video at different bit rates which are shown in table 20 and table 21.

<table>
<thead>
<tr>
<th>Format Technique</th>
<th>Mobcal Average</th>
<th>Mobcal CI 95%</th>
<th>Parkrun Average</th>
<th>Parkrun CI 95%</th>
<th>Ducks Take Off Average</th>
<th>Ducks Take Off CI 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-BC</td>
<td>41.4</td>
<td>5.1</td>
<td>16.2</td>
<td>3.7</td>
<td>29.8</td>
<td>6.6</td>
</tr>
<tr>
<td>SS-NN</td>
<td>40.0</td>
<td>5.4</td>
<td>16.2</td>
<td>3.9</td>
<td>25.0</td>
<td>5.9</td>
</tr>
<tr>
<td>TS-RF</td>
<td>54.4</td>
<td>7.5</td>
<td><strong>27.2</strong></td>
<td>6.0</td>
<td>27.5</td>
<td>5.1</td>
</tr>
<tr>
<td>NOR</td>
<td><strong>61.9</strong></td>
<td>7.0</td>
<td>24.1</td>
<td>4.7</td>
<td><strong>30.6</strong></td>
<td>5.1</td>
</tr>
</tbody>
</table>

Table 20 Average and CI of 95% for each formatting technique with respect to 300kbps bit rate test sequences.

<table>
<thead>
<tr>
<th>Format Technique</th>
<th>Mobcal Average</th>
<th>Mobcal CI 95%</th>
<th>Parkrun Average</th>
<th>Parkrun CI 95%</th>
<th>Ducks Take Off Average</th>
<th>Ducks Take Off CI 95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS-BC</td>
<td>52.0</td>
<td>5.9</td>
<td><strong>35.1</strong></td>
<td>5.2</td>
<td><strong>37.9</strong></td>
<td>5.4</td>
</tr>
<tr>
<td>SS-NN</td>
<td>46.2</td>
<td>6.2</td>
<td>33.0</td>
<td>5.0</td>
<td>36.7</td>
<td>5.5</td>
</tr>
<tr>
<td>TS-RF</td>
<td>59.1</td>
<td>6.6</td>
<td>28.0</td>
<td>5.8</td>
<td>33.1</td>
<td>5.7</td>
</tr>
<tr>
<td>NOR</td>
<td><strong>70.5</strong></td>
<td>5.9</td>
<td>25.9</td>
<td>4.9</td>
<td>33.1</td>
<td>5.5</td>
</tr>
</tbody>
</table>

Table 21 Average and CI of 95% for each formatting technique with respect to 600kbps bit rate test sequences.

From these two tables it can be seen that for the lower bit rate the normal technique is preferred except for Parkrun video, where the frame rate method seems to have small advantage. However, for 600kbps the Bicubic interpolation method is preferred for two of the videos.
We have concluded the following relation for calculating the assessments for 32 subject grades to obtain the quality attained for the proposed video formatting techniques when compared to the normal format.

\[ \mathcal{V}Quality_i = \sum_{j=1}^{32} \left( \frac{Quality_{ij} - Normal_{ij}}{Normal_{ij}} \right) \times 100 \quad \text{----------(1)} \]

\( i=1, 2, 3, 4 \ldots 12 \)
\( j=1, 2, 3, 4 \ldots 32 \)

Where, \( Quality_{ij} \) and \( Normal_{ij} \) are the quality of the proposed video formats and normal video respectively. \( \mathcal{V}Quality_i \) represents the quality attained for a video sequence when compared with normal video format, in percentage. ‘\( i \)’ represents types of generated video format and ‘\( j \)’ represents the subjects.

By using equation (1), we have calculated the \( \mathcal{V}Quality_i \) for all the video formats and stacked for analysis. \( \mathcal{V}Quality_i \) for normal video format is 0% since it is compared to itself. Comparing formatted videos with the normal video gives the quality achieved by each technique. Table 22 and table 23 gives the percentage level of each formatted video/scaled video compared to the normal video/un-scaled video. Here the (–ve) sign represents the degradation of quality compared to normal video format, similarly (+ve) represents the improved quality of video when compared to normal video format. The analysis of each video scaling technique at 300kbps and 600kbps are as follows.

<table>
<thead>
<tr>
<th>Video formats</th>
<th>Mobcal</th>
<th>Parkrun</th>
<th>Ducks Take Off</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS- Bicubic Interpolation method</td>
<td>-33.2%</td>
<td>-33.0%</td>
<td>-2.7%</td>
</tr>
<tr>
<td>SS- Nearest Neighbor</td>
<td>-35.4%</td>
<td>-32.9%</td>
<td>-18.7%</td>
</tr>
<tr>
<td>TS- Repeat Frame method</td>
<td>-12.2%</td>
<td>+12.8%</td>
<td>-10.1%</td>
</tr>
</tbody>
</table>

Table 22 Video quality level of scaled video formats compared to normal video format for 300kbps.

<table>
<thead>
<tr>
<th>Video formats</th>
<th>Mobcal</th>
<th>Parkrun</th>
<th>Ducks Take Off</th>
</tr>
</thead>
<tbody>
<tr>
<td>SS- Bicubic Interpolation method</td>
<td>-26.3%</td>
<td>+35.3%</td>
<td>+14.4%</td>
</tr>
<tr>
<td>SS- Nearest Neighbor</td>
<td>-35.4%</td>
<td>+27.2%</td>
<td>+10.9%</td>
</tr>
<tr>
<td>TS- Repeat Frame method</td>
<td>-16.1%</td>
<td>+8.1%</td>
<td>+0.0%</td>
</tr>
</tbody>
</table>

Table 23 Video quality level of scaled video formats compared to normal video format for 600kbps.

From the above results in table 22, we have concluded that proposed video formats encoded at 300kbps bit rate are not effective to attain better quality when compared to the normal video format. From the above calculations, the video formats Mobcal are represented with -35.4% to -12.2% of quality degradation compared to normal video quality. Qualities of Parkrun video formats are resulting -33.1% to
+12.8%, similarly, Ducks take off are ranging from -2.7% to -10.1% respectively. From all the test sequences generated at 300kbps the video format with TS-Repeat frame method for Parkrun is resulting high i.e. +12%.

Quality of encoded video format at 600kbps was getting better response from the subject, when compared with normal video format shown in table 23. The Mobcal video format was remaining with degradation of quality in the range of -16.1% to -35.4%. Parkrun video format was resulting with the improvement of +8.1% to +35.3% of normal video quality. Similarly, Ducks take off was resulting with the positive response of 0.0% to 14.4% of normal video quality. Graphical representation of these quality measurements are shown in figure 29 and figure 30.

Figure 29 Subjective quality difference of video formats coded at 300kbps compared to normal video format.
Figure 30 Subjective quality difference of videos formats coded at 600kbps compared to normal video format.

The graphs illustrate that for 300kbps only one of the 9 sequences shows an improvement by any of the scaling methods, while for 600kbps more than half of the sequences shows improvement.

6.3. Conclusion

In this thesis work, we have done experiments to test four different methods of video compression to determine which gives best visual quality under bit rate restrictions. Besides regular compression which we name normal compression in this report, two methods of spatial scaling and one method of temporal scaling were applied. The methods are SS-Bicubic interpolation, SS-Nearest neighbor, TS-Repeat frame method and Normal format for three sequences at two different bit rates. During the encoding process, it was discovered that these methods of spatial and temporal scaling reduce the net encoding time considerably, mainly because there is lesser data to be encoded as compared to the normal format. A general trend of better visual quality was seen in the case of normal compression without any scaling for the three video sequences encoded at 300 kbps. However, the two kinds of scaling prior to compression resulted in some videos with better subjective quality than the normal compression at the bit rate of 600 kbps. This improvement in quality was in the videos called Parkrun and Ducks take off that implies it is better to scale low motion content videos before encoding to achieve higher visual quality and lesser encoding process time. It is being inferred that the scaling will improve the visual quality even further for bitrates higher than 600 kbps, for low textured and low motion dynamics videos. The scaling does not give the same increased quality for the video Mobcal at 600 kbps, which may be due to the complex scene and color richness. In general, the
perceptual quality of videos was found better at 600kbps than 300 kbps, as one can expect.

Future work should focus on testing these scaling techniques for other frame sized videos and for higher bit rates.
## APPENDIX A

### List of Acronyms used

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BC</td>
<td>Bicbic Interpolation</td>
</tr>
<tr>
<td>BER</td>
<td>Bit Error Rate</td>
</tr>
<tr>
<td>DSCQS</td>
<td>Double Stimulus Continuous Quality Scale</td>
</tr>
<tr>
<td>DSIS</td>
<td>Double Stimulus Impairment Scale</td>
</tr>
<tr>
<td>DSR</td>
<td>Dynamic Search Range</td>
</tr>
<tr>
<td>Fps</td>
<td>Frames per second</td>
</tr>
<tr>
<td>FR</td>
<td>Frame Rate</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HVS</td>
<td>Human Visual Systems</td>
</tr>
<tr>
<td>IDC</td>
<td>Indication Level of Codec</td>
</tr>
<tr>
<td>IDE</td>
<td>Integrated Development Environment</td>
</tr>
<tr>
<td>ME</td>
<td>Motion Estimation</td>
</tr>
<tr>
<td>MOS</td>
<td>Mean Opinion Score</td>
</tr>
<tr>
<td>MSE</td>
<td>Mean Squared Error</td>
</tr>
<tr>
<td>NN</td>
<td>Nearest Neighbor</td>
</tr>
<tr>
<td>PVQA</td>
<td>Perceptual Video Quality Assessment</td>
</tr>
<tr>
<td>QP</td>
<td>Quantization Parameters</td>
</tr>
<tr>
<td>QVGA</td>
<td>Quarter Video Graphic Array (320 x 240)</td>
</tr>
<tr>
<td>RF</td>
<td>Repeat Frame</td>
</tr>
<tr>
<td>SAMVIQ</td>
<td>Subjective Assessment Methodology for Video Quality</td>
</tr>
<tr>
<td>SDSCE</td>
<td>Simultaneous Double Stimulus for Continuous Evaluation</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal to Noise Ratio</td>
</tr>
<tr>
<td>SS</td>
<td>Spatial Scaling</td>
</tr>
<tr>
<td>SSCQE</td>
<td>Single Stimulus Continuous Quality Evaluation</td>
</tr>
<tr>
<td>TS</td>
<td>Temporal Scaling</td>
</tr>
<tr>
<td>VGA</td>
<td>Video Graphic Array (640 x 480)</td>
</tr>
</tbody>
</table>
APPENDIX B

This section provides the command lines used in FFMPEG, Virtual Dub and JM reference software for the generation of the test video sequences in our thesis work:

Commands used in FFMPEG [19]

- Converting raw YUV video codec to raw AVI container:

By using this command in FFMPEG the raw video of YUV format with 1280x720 with pixel format 4:2:0 is converted to raw codec of frame size 640x480 VGA in AVI format.

```
ffmpeg -s 1280x720 -pix_fmt YUV420p -i input.YUV -vcodec rawvideo -s 640x480 -pix_fmt YUV420p output.avi
```

- Converting raw AVI VGA files to QVGA format:

Using this command, we can compress the frame size from VGA to QVGA within AVI container.

```
ffmpeg -s 640x480 -pix_fmt YUV420p -i input1.avi -s 320x240 -pix_fmt YUV420p output1.avi
```

- Converting raw AVI to YUV container:

For VGA format:

This is the command for converting the VGA format AVI file to VGA format YUV container.

```
ffmpeg -s 640x480 -pix_fmt YUV420p -i input2.avi -s 640x480 -pix_fmt YUV420p output2.YUV
```

For QVGA format:

Similarly, for converting the QVGA format AVI file to QVGA format YUV container.

```
ffmpeg -s 320x240 -pix_fmt YUV420p -i input2.avi -s 320x240 -pix_fmt YUV420p output2.YUV
```
Syntax used in Virtual dub

- For resizing the video file in Virtual Dub is:

Initially the video has to be opened in Virtual Dub, e.g. a QVGA size is opened and the target is to resize to VGA using Bilinear Interpolation technique. The process for this is

*Video-* > *Filters-* > *add-* > *resize*

Here we can give the required frame size (e.g. 620x480) and specify the technique from the dropdown list (e.g. Bilinear Interpolation). By doing this we can resize the video sequence from QVGA to VGA.

- For down scaling the video in temporal format:

We can down scale the video in temporal format i.e. if a video with frame rate: 30fps is loaded in Virtual Dub, it can be converted to 15fps using the following options

*Video-* > *Frame rate control-* > *Frame rate conversion*

Here by selecting the radio button *Process every other frame (decimate by 2)* where every second frame is removed by this the 30fps video is converted to 15fps.

Similarly, if the input video is of 15fps it can be converted to 30fps by selecting the radio button *Convert to fps: 30*, here every successive frames are repeated to attain final output.

Coding parameters used in JM reference software:

```
# The following parameters have been modified for each sequence in the config file

InputFile = "input.YUV"    # Input sequence
FramesToBeEncoded = 250    # Number of frames to be coded
FrameRate = 12.5    # Frame Rate per second (0.1-100.0)
SourceWidth = 640# Source frame width
SourceHeight = 480 # Source frame height
OutputWidth = 640 # Output frame width
OutputHeight = 480 # Output frame height
TraceFile = "input_300kbps.txt" # Trace file
ReconFile = "input_300kbps_rec.YUV" # Reconstruction YUV file
OutputFile = "input_300kbps_264.264" # Bitstream
StatsFile = "input_300kbps_stats.dat" # Coding statistics file
ProfileIDC = 66    # Profile IDC (66=baseline, 77=main, 88=extended
FREXT Profiles: 100=High, 110=High 10, 122=High 4:2:2, 244=High 4:4:4, 44=CAVLC 4:4:4 Intra)
```
# Rate control

RateControlEnable = 1  # 0 Disable, 1 Enable
Bitrate = 300000 # Bitrate (bps)

# Fast Motion Estimation Control Parameters

SearchMode = 1  # Motion estimation mode
# -1 = Full Search
# 0 = Fast Full Search (default)
# 1 = UMHexagon Search
# 2 = Simplified UMHexagon Search
# 3 = Enhanced Predictive Zonal Search (EPZS)
Reference:


[3]. Huey-Min Sun; Yan-Kai Hung; “Comparing Subjective Perceived Quality with Objective Video Quality by Content Characteristics and Bit Rates”, IEEE NISS’09; pp:624-629; Sept 2009

[4]. Ofer Hadar; Ron Shmueli; RHM Huber; “Effects of compression parameters on perceived quality of video stream over a lossy internet protocol network”, Available from: http://www.cse.bgu.ac.il/hadar/Pdf/26%20-%20JOE087003.pdf [verified November 2011]


[6]. Yunayi Xue; Yen Fu Ou; “Perceptual Video Quality Assessment on a Mobile Platform Considering both Spatial Resolution and Quantization Artifacts”, IEEE PA’2010; pp: 201-208; Feb. 2011.

[7]. Yunayi Xue; Yen Fu Ou; “Perceptual Quality Assessment of Video considering both Frame Rate and Quantization Artifacts”, IEEE Circuits and Systems for Video Technology; pp: 286-298; March 2011.


[14]. Ulrich Riter; and Jari Korhonen; “Comparing Apples and Oranges: Subjective Quality Assessment of Streamed Video with Different Types of Distortion”, IEEE QoMEx’09; p,p:127-132; Sept’09.


[18]. Toshiko Tominaga; Takanori Hayashi; Jun Okamoto; and Akira Takahashi; “Performance Comparisons of Subjective Quality Assessment Methods for Mobile Video”, IEEE QoMEX’10; p.p:82-87; July 2010.

