

Quality Adapted Backlight Scaling (QABS) for Video Streaming to Mobile Handheld Devices

Liang Cheng*, Stefano Bossi⁺, Shivajit Mohapatra*,
Magda El Zarki*, Nalini Venkatasubramanian*, and Nikil Dutt*
Donald Bren School of Information and Computer Science
University of California, Irvine, CA 92697

*{lcheng61, mopy, magda, nalini, dutt}@ics.uci.edu, ⁺stboss@tin.it

ABSTRACT

For a typical portable handheld device, the backlight accounts for a significant percentage of the total energy consumption (e.g., around 30% for a Compaq iPAQ 3650). Substantial energy savings can be achieved by dynamically adapting backlight intensity levels on such low-power portable devices. In this paper, we analyze the characteristics of video streaming services and propose an adaptive scheme called Quality Adapted Backlight Scaling (QABS), to achieve backlight energy savings for video playback applications on handheld devices. Specifically, we present a fast algorithm to optimize backlight dimming while keeping the degradation in image quality to a minimum so that the overall service quality is close to a specified threshold. Additionally, we propose two effective techniques to prevent frequent backlight switching, which negatively affects user perception of video. Our initial experimental results indicate that the energy used for backlight is significantly reduced, while the desired quality is satisfied. The proposed algorithms can be realized in real time.

1 Introduction

With the widespread availability of 3G cellular networks, mobile hand-held devices are increasingly being used to provide streaming video content. These devices have stringent power constraints because they use batteries with finite lifetime. On the other hand, multimedia services are known to be very resource intensive and tend to exhaust battery resources quickly. Therefore, conserving power to prolong battery life is an important research problem that needs to be addressed, specifically for video streaming applications on mobile handheld devices.

Most hand-held devices are equipped with a TFT (Thin-Film Transistor) LCD (Liquid Crystal Display). For these devices, the display unit is driven by the

illumination of backlight. The backlight consumes a considerable percentage of the total energy usage of the handheld device; it consumes 20%-40% of the total system power (for Compaq iPAQ) [1].

Dynamically dimming the backlight is considered as an effective method to save energy [1, 2, 3] with scaling up of the pixel luminance to compensate for the reduced fidelity. The luminance scaling, however, tends to saturate the bright part of the picture, thereby affecting the fidelity of the video quality.

In [2], a dynamic backlight luminance scaling (DLS) scheme is proposed. Based on different scenarios, three compensation strategies are discussed, i.e., brightness compensation, image enhancement, and context processing. In particular, for the luminance compensation, the distortion function is defined in Equation (7) of their paper. However, their calculation of the distortion does not consider the fact that the clipped pixel values do not contribute equally to the quality distortion.

In [3], a similar method named concurrent brightness and contrast scaling (CBCS) technique is proposed. CBCS aims at conserving power by reducing the backlight illumination while retaining the image fidelity through preservation of the image contrast. The image distortion is defined as the loss of luminance contrast. Hence, it is proposed to compensate for the backlight dimming while maintaining the contrast, and both the undershot and overshoot region are clipped, as is seen in Equation(7) of their paper. Their distortion definition and the compensation may be good for static image based applications, such as the graphic user interface (GUI) and maps, but might not be suitable for the streaming video, because their contrast compensation further compromises the fidelity of the images. In addition, [2] and [3] do not solve the problem of the frequent backlight switching which can be quite distracting to the end user.

In this paper, we explicitly incorporate video qual-

ity into the backlight switching strategy and propose the quality adaptive backlight scaling (QABS) scheme. The backlight dimming affects the brightness of the video. Therefore, we only consider the luminance compensation such that the lost brightness can be restored. The luminance compensation, however, inevitably results in quality distortion. For the video streaming application, the quality is normally defined as the resemblance between the original and processed video. Hence, for the sake of simplicity and without loss of generality, we define the quality distortion function as the mean square error (MSE)(see Equation (1)) and the quality function as the peak signal to noise ratio (PSNR)(see Equation (2)), both of which are well accepted objective video quality measurements.

$$MSE = \frac{1}{M} \times \sum_{i=1}^M (x_i - y_i)^2 \quad (1)$$

$$PSNR(dB) = 10 \log_{10} \sum_{i=1}^M \frac{255^2}{(x_i - y_i)^2} \quad (2)$$

where x_i and y_i are the original pixel value and the reconstructed pixel value, respectively. M is the number of pixels per frame.

While MSE and PSNR are often attacked for their inability to correlate to the subjective quality under certain circumstances [4] [5], they are still widely used to assess video quality. The detailed discussion of the human visual system and the corresponding perceptual quality is beyond the scope of this paper. It is to be noted that any improved quality metrics may be adopted to replace MSE/PSNR metrics used here without affecting the validity of our proposed schemes.

As is mentioned in [3], for video applications, the continuous change in the backlight factor will introduce inter-frame brightness distortion to the observer. In our experiments, we find that the “unnecessary” backlight changes fall into two categories: (1) small continuous changes over adjacent frames; (2) abrupt huge changes over a short period. Therefore, we propose to quantize the calculated backlight to eliminate the small continuous change and use a low-pass digital filter to smooth the abrupt changes.

The rest of the paper is organized as follows. In Section 2, we introduce the principle of the LCD display - experimental results show that backlight dimming saves energy while the pixel luminance compensation results in minimal overhead. In Section 3, we present our QABS scheme, which includes finding the backlight dimming factor and two supplementary

methods to avoid excessive backlight switching. Section 4 shows our prototype implementation, experimental methodology and simulation results. We conclude our work in Section 5.

2 Characteristics of LCD

In this section, we introduce the characteristics of the LCD unit from two perspectives, the LCD display mechanism and the LCD power consumption, both of which form the basis for our system design.

2.1 LCD display

The LCD panel does not illuminate itself, but displays by filtering the light source from the back of the LCD panel [2][3]. There are three kinds of TFT LCD panels: transmissive LCD, reflective LCD, and transflective LCD. In transmissive LCD, the LCD pixels are illuminated from behind (i.e., opposite the viewer) using a backlight. The transmissive LCD offers a high quality display and is widely used in laptop personal computers. However, it is not legible with backlight turned off. The reflective LCD has a reflector on the back, which may reflect the ambient environment light or a frontlight. (Note that we will refer to both backlight and frontlight as backlight henceforth.) Compared to transmissive LCD, reflective LCD uses modest amount of power for illumination. Hence, most of the handheld devices, such as PDAs and cell phones use reflective LCD. Transflective LCD combines both transmissive and reflective mechanisms. It is not as common as the other two types.

Because the power constraint of the handheld device is especially important and since it is the most commonly used LCD, we focus in this paper on the reflective LCD, as shown in Figure 1. Henceforth, when we mention LCD, we refer to reflective LCD. Nevertheless, our idea is generic to any backlight based LCD.

The perceptual luminance intensity of the LCD display is determined by two components: backlight brightness and the pixel luminance. The pixel luminance can be adjusted by controlling the light passing through the TFT array substrate. People may detect the change of the display luminance intensity if either of these two components is adjusted. That is, the backlight brightness and the pixel luminance can compensate each other. In Section 2.2, we will show that the pixel luminance does not have a noticeable impact on the energy consumption, whereas the backlight illumination results in high energy consumption. Hence, in general, dimming backlight level while compensating the pixel luminance is an effective way to conserve

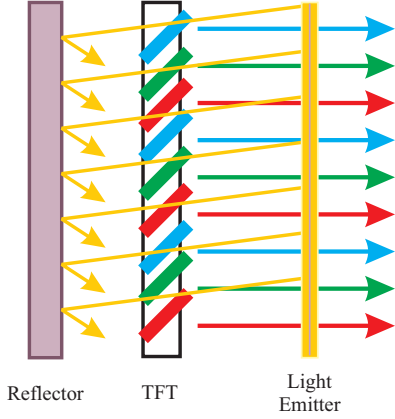


Figure 1: Reflective LCD.

battery power in hand-held devices.

Let the backlight brightness level and the pixel luminance value be L and Y , respectively, and the perceived display luminance intensity I . We may denote I using Equation (3).

$$I = \rho \times L \times Y \quad (3)$$

where ρ is the constant ratio, denoting the transmittance attribute of the LCD panel, and $\rho \times Y$ turns out to be the transmittance of the pixel luminance.

We may reduce the backlight level to L' by multiplying L with a dimming factor α , i.e., $L' = L \times \alpha$, $0 < \alpha < 1$. To maintain the overall display luminance I invariable, we need to boost the luminance of the pixel to Y' . Since the pixel luminance value is normally restricted by the number of bits that represent it (denoted as n), Y' may be clipped if the original value of Y is too high or the α is too low. The compensation of the backlight is described in Equation (4).

$$Y' = \begin{cases} Y/\alpha, & \text{if } Y < \alpha \times 2^n \\ 2^n, & \text{if } Y \geq (\alpha \times 2^n) \end{cases} \quad (4)$$

Combining Equation (4) and Equation (3), we have

$$I' = \begin{cases} I, & \text{if } Y < \alpha \times 2^n \\ \rho \times L \times \alpha \times 2^n & \text{if } Y \geq (\alpha \times 2^n) \end{cases} \quad (5)$$

Equation (5) clearly shows that the perceived display intensity may not be fully recovered, instead, it is clipped to $\rho \times L \times \alpha \times 2^n$ if $Y \geq (\alpha \times 2^n)$. In Figure 4, we illustrate the clipping effect of the display luminance.

In Figure 2, we show an image and its luminance histogram. This image is the first frame of a typical

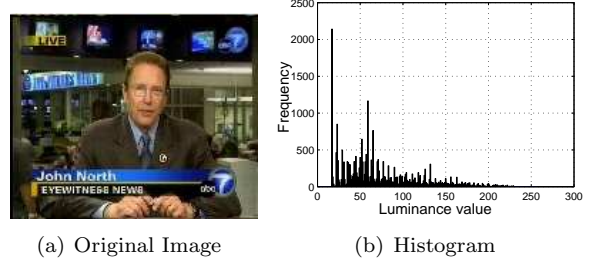


Figure 2: Original image and its luminance histogram

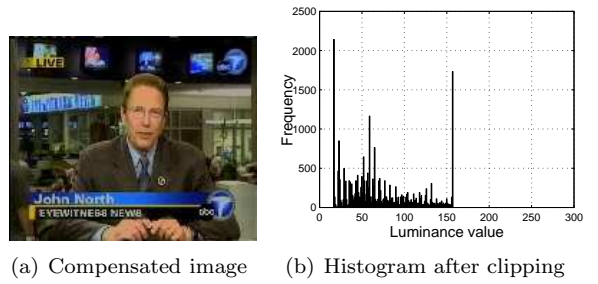


Figure 3: Compensated image and its luminance histogram

news video clip (“ABC eye witness news”) captured from the broadcasting TV signal. Figure 3 illustrates the image and its luminance histogram after the backlight dimming and the pixel luminance compensation. Figure 3-b shows that the pixels with luminance higher than 156 are all clipped to 156. This clipping effect eliminates the variety in the bright areas, which is subjectively perceived as the luminance saturation and is objectively assessed as 30dB with the reference of the original image in Figure 2-a.

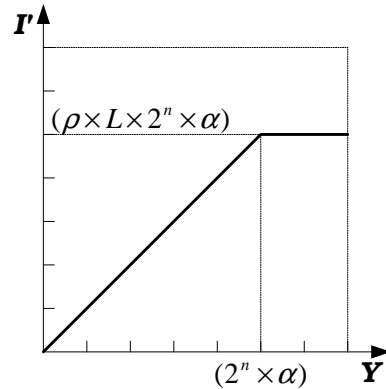


Figure 4: Clipping effect.

2.2 LCD power model

In our experiments, we observe that the backlight dimming can save energy whereas the compensation process, i.e., scaling up the luminance of the pixel, has a negligible energy overhead. We measure the energy saving as a difference of the total system power consumption with backlight set to different levels from that with the backlight turned to the maximum (brightest). Figure 5 shows the plot between the various backlight levels and their corresponding energy consumption for a Compaq iPAQ 3650 running Linux. A more detailed setup of our experiments is described in Section 4. It is noticed that the backlight energy saving is almost linear to the backlight level and can be estimated using Equation (6).

$$y = a1 \times x + a2 \quad (6)$$

where y is the energy savings in Watt; x denotes the backlight level; $a1$ and $a2$ are coefficients. We apply the curve fitting function of MATLAB and obtain $a1 = -0.0029567$ and $a2 = 0.73757$ with the largest residual fitting error as 0.085731.

Contrary to the backlight switching, the pixel luminance scaling is uncorrelated to the energy consumption. In Figure 6, we show that for one specified backlight level (BL) the system energy consumption basically remains stable and is independent of the luminance scaling.

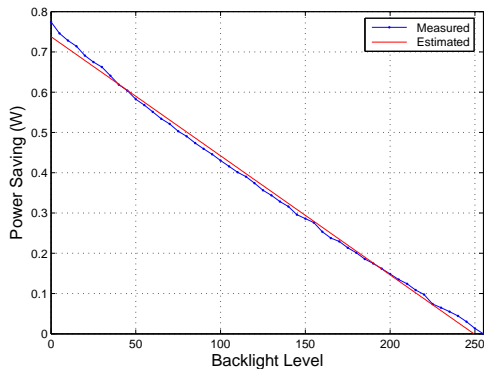


Figure 5: Power saving versus backlight level.

Figure 5 and Figure 6 justify the validity of the generic backlight power conservation approach, i.e., dimming the backlight while enhancing the pixel luminance value. In the next section, we apply this method to the video streaming scenario, discussing a practical scheme to optimize the backlight dimming while taking into consideration the effect on video distortion.

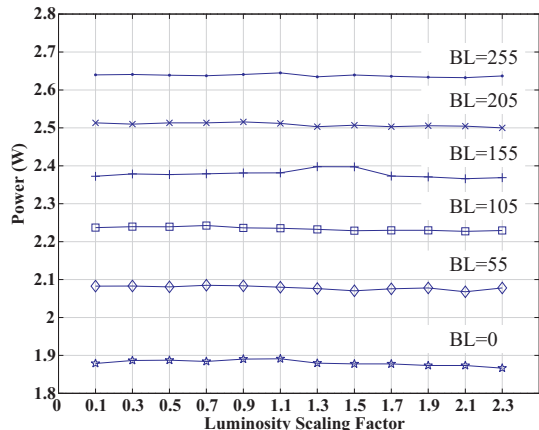


Figure 6: Energy overhead for the luminance scaling. “BL” refers to the backlight level; “Luminosity Scaling Factor” = α .

3 Adaptive Backlight Scaling

As explained in Equation (5), the backlight scaling with the luminance compensation may result in quality distortion. The amount of backlight dimming, therefore, has to be restricted such that the video fidelity will not be seriously affected.

3.1 Optimized Backlight Dimming

We define the optimized backlight dimming factor as the one whose induced distortion is closest to a specified threshold. Henceforth, we replace the factor α with the real backlight level $Alfa$, $Alfa = N \times \alpha$ (N is the number of backlight levels (256 for Linux on iPAQ)), and the optimized backlight dimming is represented as $Alfa^*$.

In Figure 7, we illustrate the image quality distortion in terms of MSE over different backlight levels. (Note that we use the image shown in Figure 2-a.) We see that as $Alfa$ increases, the induced video quality distortion due to the brightness saturation monotonously decreases. Hence, for a given distortion threshold, we can find a unique $Alfa (= Alfa^*)$ for each image. In video applications, for a given distortion, different frames may have distinct $Alfa^*$, depending on the luminance histogram of that frame. However, it is hard to have an accurate analytical representation of the quality distortion using $Alfa$ as a parameter. We therefore adopt an optimized search based approach, where we calculate the MSE distortion with different $Alfa$ until the specified distortion threshold is met. The results of our scheme are accurate and can be used as the benchmark for the design of other

analytical methods.

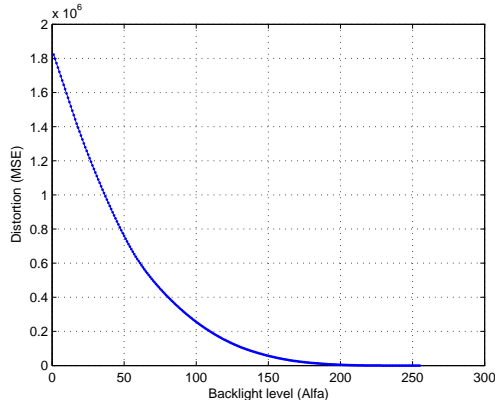


Figure 7: MSE with different $Alfa$

Figure 8 shows the exhaustive searching algorithm of finding $Alfa^*$ for one image. The call $FindAlfa(th)$ takes the distortion threshold (th) as input, and returns the $Alfa^*$ as output. Note that $MSE(Alfa)$ calculates the MSE with the specified $Alfa$ for one frame.

```

Proc: FindAlfa(th)
1: Alfa := 0;
2: while Alfa ≤ N do
3:   if MSE(Alfa) > th then
4:     Alfa := Alfa + 1;
5:   else
6:     return(Alfa);
7:   end if
8: end while

```

Figure 8: Exhaustive algorithm of finding $Alfa^*$.

However, the complexity of an exhaustive search shown in Figure 8 is too high. As shown in Equation (2), the per-frame MSE calculation consists of M multiplications and $2M$ additions. M is the number of pixels in one frame, e.g., $M = 25344$ for QCIF format video. We regard the per-frame MSE as the basic complexity measurement unit. We assume that the optimized backlight level is uniformly distributed in $[0, N]$, and thus the complexity of algorithm in Figure 8 is $O(N)$. In our test, $N = 256$. Obviously, the optimized backlight dimming factor can hardly be calculated in real-time.

Therefore, we apply a faster bisection method [6] to improve the algorithm of finding $Alfa^*$. Since we can easily find an upper bound (denoted as u) and a lower bound (denoted as d) on the backlight levels, we get as

good an approximation as we want by using bisection. We assume that $u > d$ and let ϵ be the desired precision and present the algorithm in Figure 9.

```

Proc: FastFindAlfa(th, ε)
1: u := upper bound;
2: d := lower bound;
3: while (u - d) > ε do
4:   Alfa = round((d + u)/2);
5:   if (MSE(Alfa) > th) then
6:     u = Alfa;
7:   else
8:     d = Alfa;
9:   end if
10: end while
11: return(Alfa);

```

Figure 9: Fast algorithm of finding $Alfa^*$.

By using the bisection method, we may achieve the complexity of $O(\log_2 N)$ in the worst case. For instance, for $N = 256$ and $\epsilon = 1$, we only need to calculate per-frame MSE at most eight times, which is fast enough for the real-time processing.

3.2 Backlight switching smoothing

It has been discussed in [3] that the backlight dimming factor may change significantly across consecutive frames for most video applications. The frequent switching of the backlight may introduce an inter-frame brightness distortion to the observer. Hence, it is necessary to reduce frequent backlight switching.

In our study, we observe that the calculated $Alfa^*$, although based on an individual image, does not experience huge fluctuations during a video scene, i.e., a group of frames that are characterized with similar content. Actually, the redundancy among adjacent frames constitutes the major difference between the video and the static image application and has long been utilized to achieve higher compression efficiency. Hence, the backlight switching should be smoothed out within the scene and most favorably only happen at the boundary of the video scene.

We propose two supplementary methods to smooth the acquired $Alfa^*$ in the same video scene. First, we apply a low-pass digital filter to eliminate any abrupt backlight switching that is caused by the unexpected sharp luminance change. In order to lower the system complexity and avoid the oscillation caused by the signal feedback, we choose finite impulse response (FIR) filter. The passband frequency is determined by the subjective perception of the "flicker moment" and the

frame display rate. For example, if we assume 2 seconds as the “flicker moment” and the frame display rate is 10, we would have a sampling frequency $f_s = 10\text{Hz}$ and a cutting frequency $f_c = 0.5\text{Hz}$. Second, we propose to quantize the number of backlight levels, i.e., any backlight level between two quantization values can be quantized to the closest level, by which we prevent the needless backlight switching for small luminance fluctuations during one scene. For example, for an iPAQ 3650, there are as many as 256 backlight levels. However, it is unnecessary to consider all of these levels as candidates for backlight scaling, as the power saving gains for adjacent levels is negligible. Instead, we may quantize these levels to “N” levels (e.g. we use $N=5$ in our study). We switch the backlight level only if the calculated $Alfa^*$ changes drastically enough, so that it falls into another quantized level.

4 Performance Evaluation

In this section, we introduce our prototype implementation, the methodology of our measurement and the performance of the proposed algorithm.

4.1 Prototype Implementation

We have implemented a prototype of a video streaming system that incorporates our QABS based adaptations. Figure 10 shows a high level representation of our prototype system. Our implementation of the video streaming system consists of a video server, a proxy server and a mobile client. We assume that all communication between the server and the mobile client is routed through a proxy server typically located in proximity to the client.

The video server is responsible for streaming compressed video to the client using RTP [7] over UDP [8]; The proxy server transcodes the received stream, adds the appropriate control information, and relays the newly formed stream to the mobile client (Compaq iPAQ 3650 in our case). For the sake of simplicity and without loss of generality, in our initial prototype implementation, we use the proxy server to also double up as our video server, i.e., we use the video data stored on the proxy server for our testing purposes.

The proxy server includes four primary components - the video transcoder, the proposed QABS module, the signal multiplexor, and the communication manager. The transcoder uncompresses the original video stream and provides the pixel luminance information to the QABS module; the QABS module calculates the optimized backlight dimming factor based on the user quality preference feedback received from

the client (user). The multiplexor is used to multiplex the optimized backlight dimming information with the video stream. The communication manager is used to send this aggregated stream to the client. It is noted that after quantization, the backlight level can be represented by 3 bits, i.e., 30 bits/s for a display rate of 10 fps, which is a negligible bandwidth overhead.

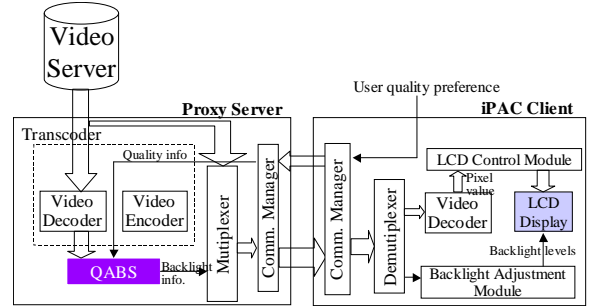


Figure 10: Prototype implementation.

On the mobile client, the demultiplexor is used to recover the original video stream and the encoded backlight information from the received stream. The LCD control module renders the decoded image onto the LCD display, which is perceptible after the backlight illumination. The backlight information is fed to the “Backlight Adjustment Module”, which concurrently sets the backlight value for the LCD. In particular, users may send the quality request to the proxy when requesting for the video, based on his/her quality preference as well as the concern of the battery lifetime.

4.2 Measurement Methodology

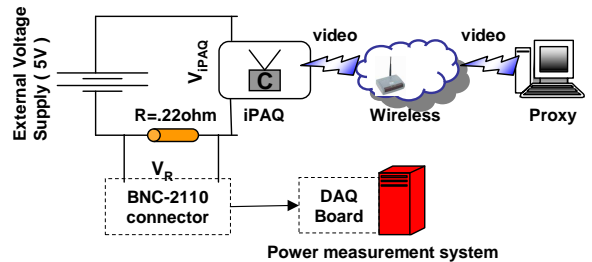


Figure 11: Setup for our measurements.

For video quality and power measurements, we use the setup shown in Figure 11. The proxy in our experiments is a Linux desktop with a 1GHz processor and 512MB of RAM. All our measurements are made on a Compaq iPAQ 3650 (running the Linux 2.4.18-rmk3 kernel), with a 200Mhz Intel StrongArm processor, with 16MB ROM, 32MB SDRAM. The iPAQ 3650

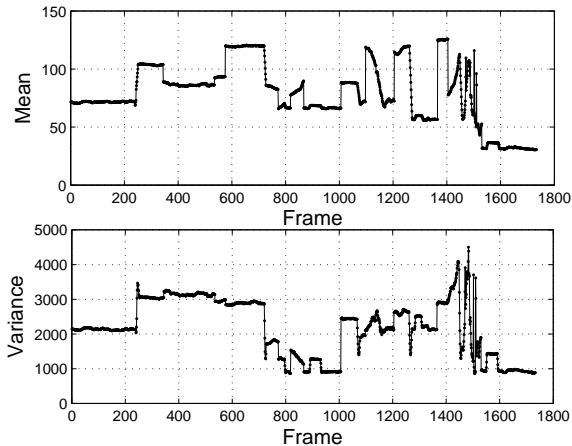


Figure 12: Basic statistics of *abc_news*.

uses a reflective LCD, whose backlight is 256-level programmable. The iPAQ uses a Cisco 350 Series Aironet 11Mbps wireless PCMCIA network interface card for communication. The batteries are removed from the iPAQ during the experiment. We use a National Instruments PCI DAQ board to sample voltage drops across a resistor and the iPAQ, and sample the voltage at 200K samples/sec. We calculate the instantaneous and average power consumption of the iPAQ using the formula $P_{iPAQ} = \frac{V_R}{R} \times V_{iPAQ}$ (Figure 11).

4.3 Experimental Results

In our simulation, we use a video sequence captured from a broadcasted “news” program, whose first frame is shown in Figure 2-a. We choose this video as the representative of a typical usage of a PDA. This sequence is composed of a set of scenes, which can be roughly characterized by Figure 12. In Figure 12, we show the basic statistics (i.e., the mean and the variance of luminance per frame) of *ABC_news*. The capturing rate is 10 frame per second, and the whole sequence lasts 1750 frames.

We assume that the users are given three quality options, fair, good, and excellent, which respectively correspond to the PSNR value of 30dB, 35dB, and 40dB. After applying the algorithm “Proc: FastFind-Alfa”, we obtain the adapted *Alfa** for these three quality preferences, as is shown in Figure 13. It can be seen that higher video quality needs higher backlight level on average.

In Figure 14, we show *Alfa** before and after the backlight smoothing process. It is seen that the small variation and the abrupt change of the backlight switching are significantly eliminated after the filtering and quantization. In addition, as we expected, the

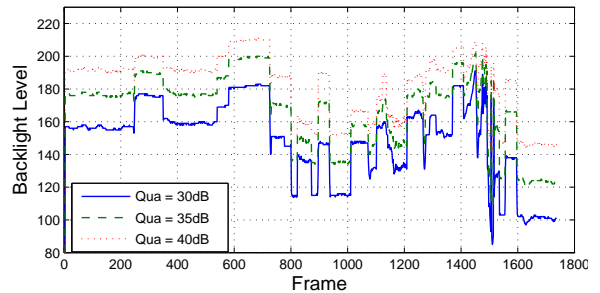


Figure 13: *Alfa** adapted to three given quality thresholds.

backlight switching mostly happens at the boundary of major scenes.

In Table 1, we summarize the results of our QABS. The mean *Alfa** of different quality preferences produce the quality on average very close to the predetermined quality threshold. It is noted that different quality requirements result in various power saving gains. Higher quality preference must be traded using more backlight energy. Nevertheless, we can still save 29% energy that is supposed to be consumed by the backlight unit if we set the quality preference to be “Excellent”. Note that the “Excellent” quality, i.e., 40dB in PSNR metric, means nearly unnoticeable quality distortion.

Table 1: Results of QABS

	Alfa Mean	Quality Mean	Power Saving
Fair	149	30.17	41.8%
Good	162	34.28	36.7%
Excellent	186	42.31	27.3%

In Figure 15, we show that the filtering and quantization process may lead to instantaneous quality fluctuation, which is contrasted to the consistent quality before backlight smoothing. Nevertheless, we observe that the quality fluctuation is around the designated quality threshold and mostly happens at scene changes. In follow up studies, we will investigate the perceptual influence of these factors, i.e., the luminance saturation and frequent backlight changes.

5 Conclusion and Future Work

In this paper, we apply a backlight scaling technique to video streaming applications, and explicitly associate backlight switching to the perceptual video quality in

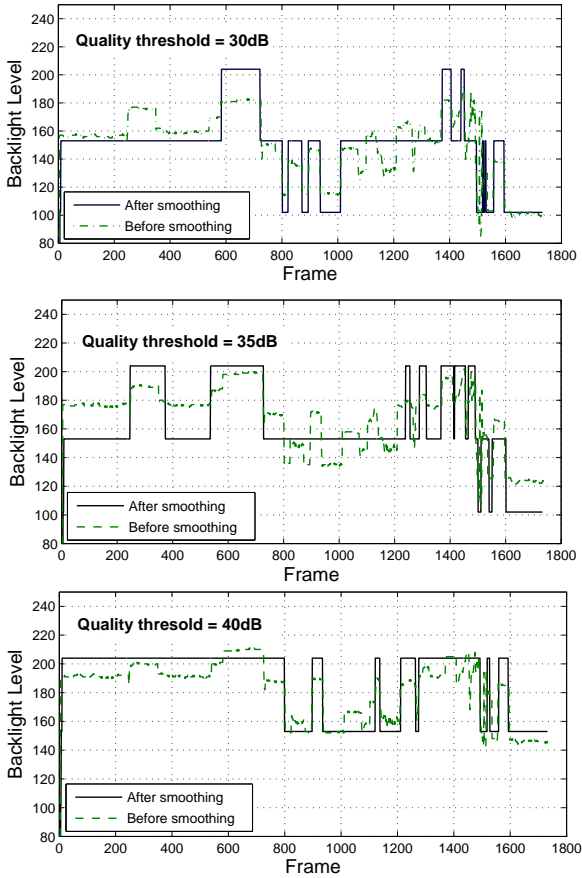


Figure 14: $Alfa^*$ before and after filtering and quantization.

terms of PSNR. The proposed adaptive algorithm is fast and effective for reducing the energy consumption while maintaining the designated video quality. To reduce the frequency of backlight changing, we propose two supplementary schemes that smooth the backlight change process such that the user perception of the video stream can be substantially improved. In addition, our proposed algorithms are tested on a prototype system.

In our future work, we plan to work on several ideas that can potentially improve our current approach. We enumerate some of these areas below: (1) We find that PSNR and MSE are not satisfactory indicators for determining the distortion caused by the brightness saturation, and they should be replaced by the metric that incorporates luminance contrast; (2) the low-pass filter is essentially heuristic and should be substituted by a more accurate scene detection technique; (3) the video decoding process in the proxy costs extra overhead, which may be eliminated if we can explore a method to conduct QABS within the compression domain. Ad-

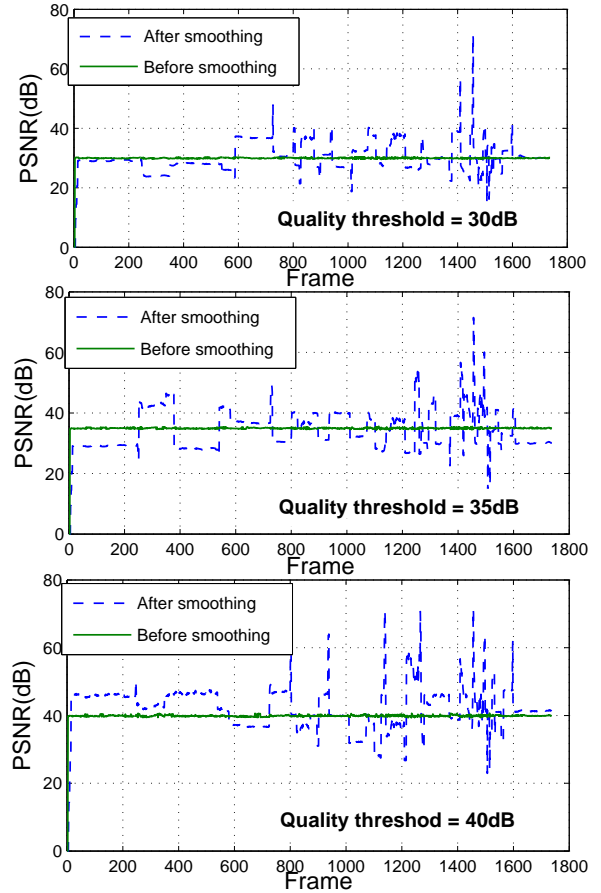


Figure 15: Quality before and after Alfa smoothing.

ditionally, we plan to incorporate the backlight saving algorithm to in overall system power management framework and design a unified cross-layer power solution for hand-held devices with LCD displays.

6 Acknowledgement

We would like to thank Michael Philpott, who helped us with the experiment setup and the power measurement.

References

- [1] S. Pasricha, M Luthra, S. Mohapatra, N. Dutt, N. Venkatasubramanian, "Dynamic Backlight Adaptation for Low Power Handheld Devices," *To appear in IEEE Design and Test (IEEE D&T), Special Issue on Embedded Systems for Real Time Embedded Systems*, Sep. 2004.

- [2] N. Chang, I. Choi, and H. Shim, "DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display," *IEEE Transaction on VLSI System*, vol. 1, Aug. 2004.
- [3] W.-C. Cheng, Y. Hou, and M. Pedram, "Power Minimization in a Backlit TFT-LCD Display by Concurrent Brightness and Contrast Scaling," *Proceedings of the Design, Automation and Test in Europe*, Feb. 2004.
- [4] S. Wolf and M. H. Pinson, "Spatial-Temporal Distortion Metrics for In-Service Quality Monitoring of Any Digital Video System," *Proc. of SPIE Int. Symp. on Voice, Video, and Data Commun.*, Sep. 1999.
- [5] M. H. Pinson and S. Wolf, "A New Standardized Method for Objectively Measuring Video Quality," *IEEE Transactions on Broadcasting*, no. 3, Sep. 2004.
- [6] J. L. Zachary, *Introduction to Scientific Programming: Computational Problem Solving Using Maple and C*. Telos Publishers, 1996.
- [7] H. Schulzrinne, S. Casner, R. Frederderick, and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications," *IETF RFC 1889*, Jan. 1996.
- [8] Internet RFC/STD/FYI/BCP Archives, "User Datagram Protocol," *IETF RFC 768*, Aug. 1980.