Hindawi Publishing Corporation
Advances in Multimedia
Volume 2007, Article ID 83715, 10 pages
doi:10.1155/2007/83715

Research Article

Quality-Based Backlight Optimization for Video Playback on Handheld Devices

Liang Cheng,[1] Shivajit Mohapatra,[2] Magda El Zarki,[3] Nikil Dutt,[3] and Nalini Venkatasubramanian[3]

[1] *NVIDIA Corporation, Santa Clara, CA 95050, USA*
[2] *Motorola Labs, Schaumburg, IL 60196, USA*
[3] *Donald Bren School of Information and Computer Science, University of California, Irvine, CA 92697-3425, USA*

Received 18 December 2006; Accepted 23 April 2007

Recommended by Haohong Wang

For a typical 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 a cross-layer optimization 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/4G cellular networks, mobile handheld devices are increasingly being designed to support 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.

Many handheld-device-based power saving techniques have been reported in the literature. They attempt to reduce power consumption at various computational levels. In [1], a number of architectural and software compiling strategies were proposed to optimize system cache and external memory access. Reference [2] specifically aims at MPEG-based applications. It proposes to scale the processor voltage and frequency to provide the necessary computing capability for decoding each video frame. In [3, 4], the power consumption of network interfaces (NICs) are optimized.

In this paper, we focus on the power consumption of the display unit in the handheld device. Most handheld devices are equipped with a thin-film transistor (TFT) liquid crystal display (LCD). For these devices, the display unit is driven by backlight illumination. The backlight consumes a considerable percentage of the total energy usage of the handheld device—for example, for a Compaq iPAQ device, it consumes 20%–40% of the total system power [5].

Dynamically dimming the backlight is considered an effective method to save energy [5–7]. The resultant reduced fidelity can be compensated for with scaling up of the pixel luminance. The luminance scaling, however, tends to saturate the bright part of the picture, thereby affecting the fidelity of the video quality.

In [6], a dynamic backlight luminance scaling (DLS) scheme is proposed. Based on different scenarios, three compensation strategies are discussed, that is, brightness compensation, image enhancement, and context processing. 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 [7], a similar method, namely, concurrent brightness and contrast scaling (CBCS), is proposed. CBCS aims at conserving power by reducing the backlight illumination while retaining the image fidelity through preservation of the image contrast. Their distortion definition and proposed compensation technique may be good for static image-based applications, such as the graphic user interface (GUI) and maps, but might not be suitable for streaming video scenarios, because their contrast compensation further compromises the fidelity of the images. In addition, neither [6] nor [7] solves the problem associated with frequent backlight switching which can be quite distracting to the end user.

In this paper, we explicitly incorporate video quality into the backlight switching strategy and propose a 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) (1) and the quality function as the peak signal to noise ratio (PSNR) (2), both of which are well-accepted objective video quality measurements,

$$\mathrm{MSE} = \frac{1}{M} \times \sum_{i=1}^{M} \left(x_i - y_i\right)^2, \tag{1}$$

$$\mathrm{PSNR(dB)} = 10 \log_{10} \sum_{i=1}^{M} \frac{255^2}{\left(x_i - y_i\right)^2}, \tag{2}$$

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.

It is well known that MSE and PSNR are not the best measures to assess perceptual quality for most video sequences [8, 9]. However, they are widely used due to their simple implementation. A 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 quality metric may be adopted to replace the used MSE and PSNR measures without affecting the validity of our proposed schemes.

As is mentioned in [7], 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 determining the backlight dimming factor and two supplementary methods to avoid excessive backlight switching. Section 4 describes our prototype implementation, experimental methodology, and simulation results. We conclude our work in Section 5.

2. CHARACTERISTICS OF LCD

In this section, we outline 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; it displays by filtering the light source from the back of the LCD panel [6, 7]. There are three kinds of TFT LCD panels: transmissive LCD, reflective LCD, and transflective LCD. In transmissive LCD, the pixels are illuminated from behind (i.e., opposite the viewer) using a backlight. The transmissive LCD offers a high quality display with large power consumption, so it is widely used in laptop personal computers. The reflective LCD has a reflector on the back, which reflects the ambient environment light or a frontlight. Compared to a transmissive LCD, a reflective LCD uses modest amounts of power for illumination. Hence, most of the handheld devices use reflective LCD. Transflective LCD combines both transmissive and reflective mechanisms but is not as commonly used in handheld devices as the other two types.

In general, both transmissive and reflective mode LCD need artificial light source to illuminate the display. Hence, reducing the light power consumption is beneficial to all these three types of LCDs. For the sake of simplicity and without loss of generality, we use backlight to represent all types of light source—frontlight is also designated as backlight in the case of a reflective LCD. All of our proposed algorithms are applicable to both backlight and frontlight. Since the reflective mode LCD is more popular to the handheld devices, we base our algorithms and measurements on a Compaq iPAQ 3650—a PDA with reflective mode LCD. Figure 1 illustrates the schematic mechanism of the reflective mode 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. Users may detect a change in the display luminance intensity if either of these two components is adjusted. That is, the backlight brightness and the pixel luminance can compensate for 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. In general, dimming the backlight level while compensating for it with the pixel luminance is an effective way to conserve battery power in handheld devices.

We can therefore conclude that reducing backlight and increasing pixel luminance will result in power savings.

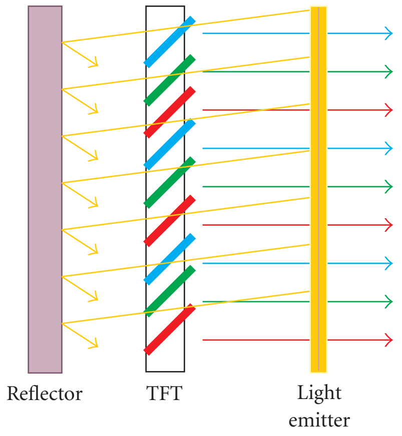


Figure 1: Reflective LCD.

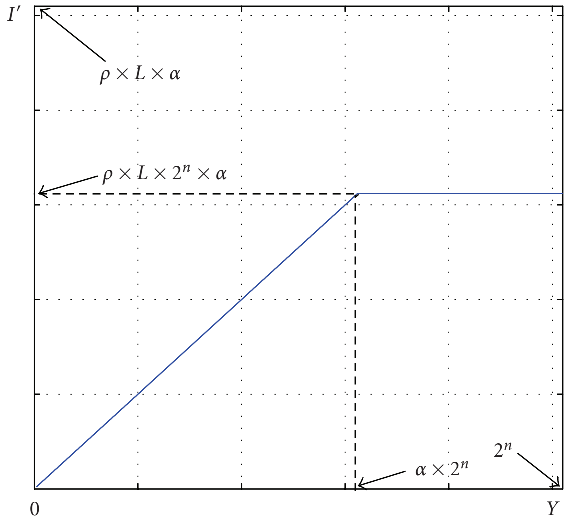


Figure 2: Clipping effect of LCD.

Let the backlight brightness level and the pixel luminance value be L and Y, respectively, and the perceived display luminance intensity be I. We may denote I using the following:,

$$I = \rho \times L \times Y, \tag{3}$$

where ρ is a constant ratio, denoting the transmittance attribute of the LCD panel, and as such $\rho \times Y$ is the transmittance of the pixel luminance.

We may reduce the backlight level to L' by multiplying L with a dimming factor α, that is, $L' = L \times \alpha$, $0 < \alpha < 1$. 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 (4),

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

Combining (4) and (3), we have

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

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 2, we illustrate the clipping effect of the display luminance.

In Figure 3, we show an image and its luminance histogram. This image is the first frame of a typical news video clip "ABC eye witness news" captured from a broadcast TV signal. Figure 4 illustrates the image and its luminance histogram after backlight dimming and pixel luminance compensation. Figure 4(b) shows that pixels with luminance higher than 156 are all clipped to 156. Compared with Figure 4(a), this clipping effect diminishes the differences in the brightness areas of the image, for example, the white caption and the face. This effect is subjectively perceived as luminance saturation and is objectively assessed as 30 dB using the PSNR quality metric with reference to the original image shown in Figure 3(a).

2.2. LCD power model

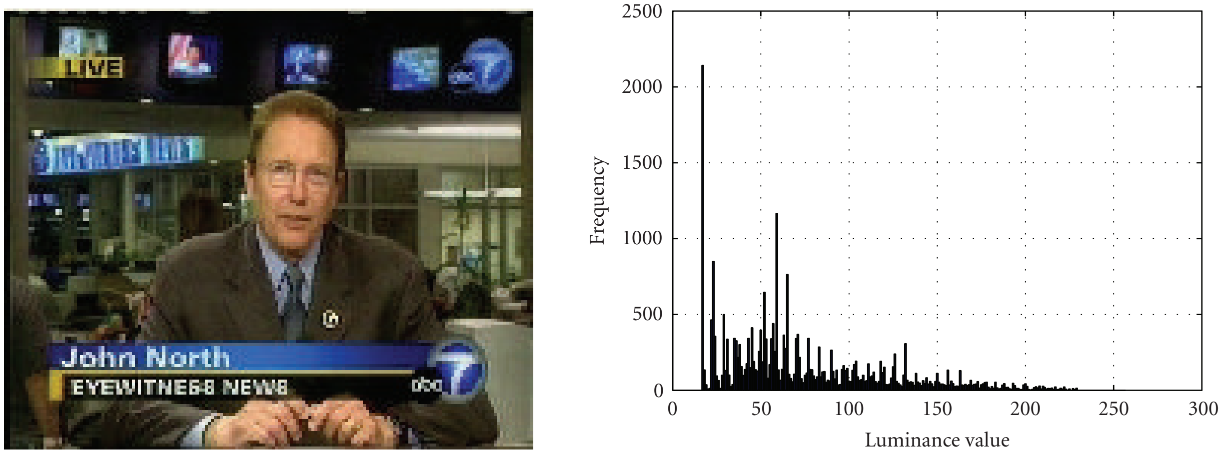
We ran several experiments and observed that dimming the backlight results in energy savings, whereas the compensation process, that is, scaling up the luminance of the pixel, has a negligible energy overhead. We measure the energy saving as the difference between the total system power consumption with the backlight set to different levels to that with the backlight turned to the maximum (brightest). In Figure 5, we plot 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 (6),

$$y = a1 \times x + a2, \tag{6}$$

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 dimming, 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.

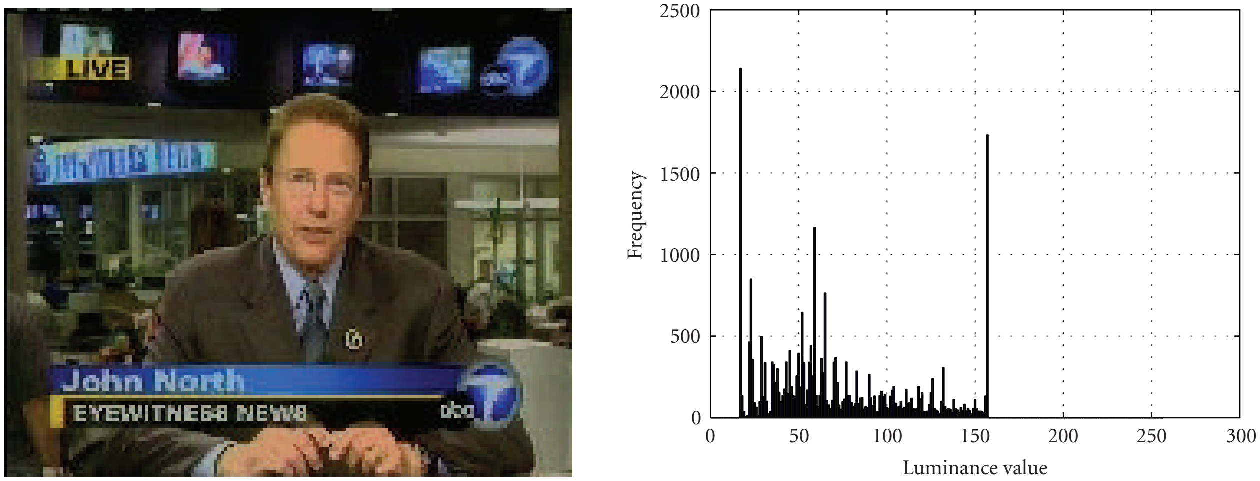
Figures 5 and 6 justify the validity of the proposed backlight power conservation approach, that is, dimming the backlight while enhancing the pixel luminance value. Note that in Figure 6,"BL" refers to the backlight level and



(a) Original image

(b) Histogram before clipping

Figure 3: Image and its luminance histogram before clipping.



(a) Compensated image

(b) Histogram after clipping

Figure 4: Image and its luminance histogram after clipping.

"luminosity scaling factor" refers to α. 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.

3. ADAPTIVE BACKLIGHT SCALING

As explained in (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 3(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

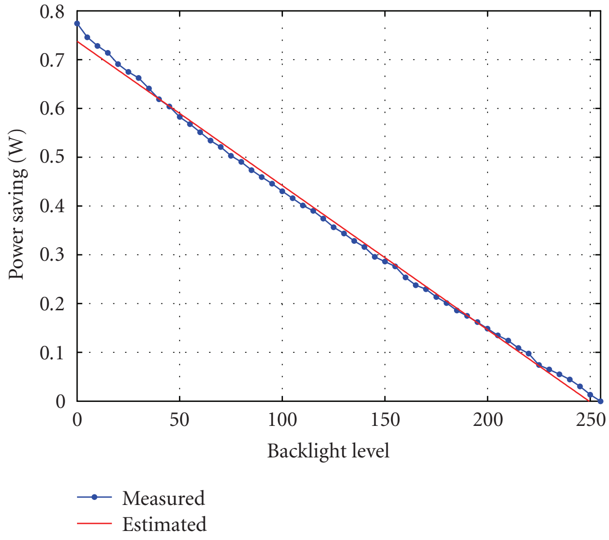


Figure 5: Power saving versus backlight level.

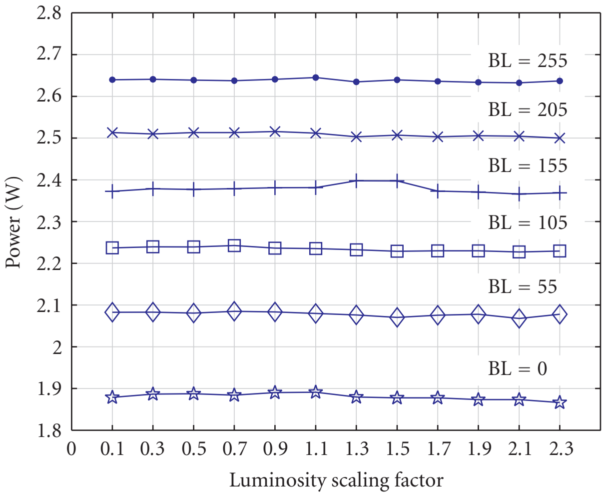


Figure 6: Energy overhead of luminosity scaling.

are accurate and can be used as the benchmark for the design of other analytical methods.

Algorithm 1 shows the exhaustive search algorithm for finding Alfa* for one image. FindAlfa(th) takes the distortion threshold (th) as input, and returns Alfa* as output. Note that MSE(Alfa) calculates the MSE with the specified Alfa for one frame.

However, the complexity of an exhaustive search shown in Algorithm 1 is too high. As shown in (2), the per-frame MSE calculation consists of M multiplications and $2M$ additions. M is the number of pixels in one frame, for example, $M = 25344$ 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 Algorithm 1 is $O(N)$. In our test, $N = 256$. It is obvious that the optimized backlight dimming factor cannot be calculated in real time.

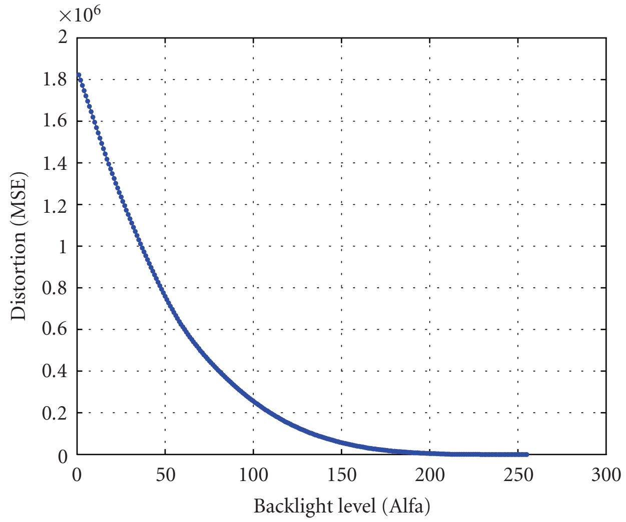


Figure 7: MSE with different Alfa.

We therefore apply a faster bisection method [10] to improve the algorithm for finding Alfa*. Since we can easily find an upper bound (denoted as u) and a lower bound (denoted as d) on the backlight level, we obtain a good approximation using this method. We assume that $u > d$ and let ϵ be the desired precision. The algorithm based on the bisection method is illustrated in Algorithm 2.

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

3.2. Smoothing the backlight switching

It has been discussed in [7] that the backlight dimming factor may change significantly across consecutive frames for most video applications. We call this abrupt backlight switching "flicker moment." The frequent switching of the backlight may also introduce an interframe brightness distortion to the observer due to the brightness compensation. 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, that is, 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 video scenes.

We propose two supplementary methods to smooth the acquired Alfa* in the same video scene. First, we apply

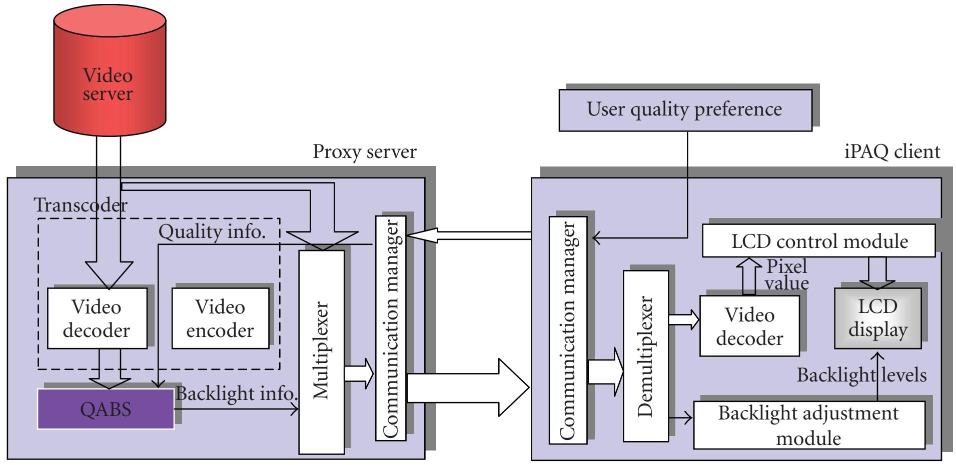


FIGURE 8: Prototype implementation.

```
Proc: FindAlfa(th)
Alfa := 0;
while Alfa ≤ N do
    if MSE(Alfa) > th then
        Alfa := Alfa + 1;
    else
        return(Alfa);
    end if
end while
```

ALGORITHM 1: Exhaustive algorithm for finding Alfa*.

```
Proc: FastFindAlfa(th,ϵ)
u := upper bound;
d := lower bound;
while (u − d) > ϵ do
    Alfa = round ((d + u)/2);
    if (MSE(Alfa) > th) then
        u = Alfa;
    else
        d = Alfa;
    end if
end while
return(Alfa);
```

ALGORITHM 2: Fast algorithm for finding Alfa*.

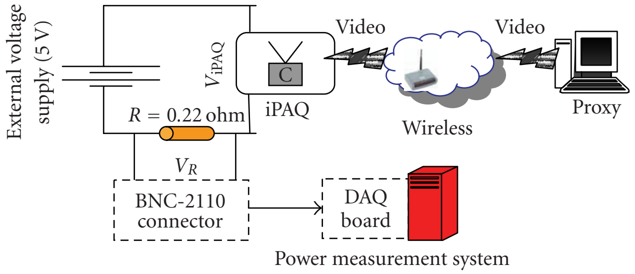


FIGURE 9: Setup for our measurements.

a low-pass digital filter to eliminate any abrupt backlight switching that is caused by the unexpected sharp luminance change. The passband frequency is determined by the subjective perception of the “flicker moment” and the frame display rate. Second, we propose to quantize the number of backlight levels, that is, 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. In our experiments, we quantize all 256 levels to “N” levels ($N = 5$ in our study). We switch the backlight level only if the calculated Alfa* changes drastically enough, and falls into another quantized level.

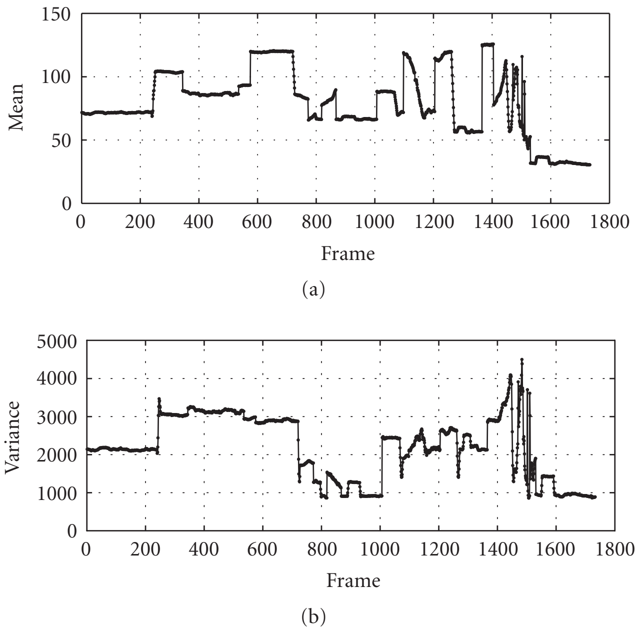
4. PERFORMANCE EVALUATION

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

4.1. Prototype implementation

Figure 8 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. 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). In our initial implementation, for the sake of simplicity and without loss of generality, we use the proxy server to also double up as our video server.



(a)

(b)

FIGURE 10: Basic statistics of *abc_news*.

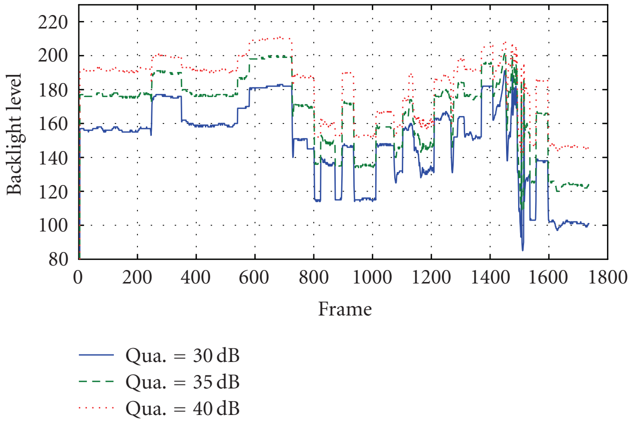


FIGURE 11: Alfa* adapted to three-given quality thresholds.

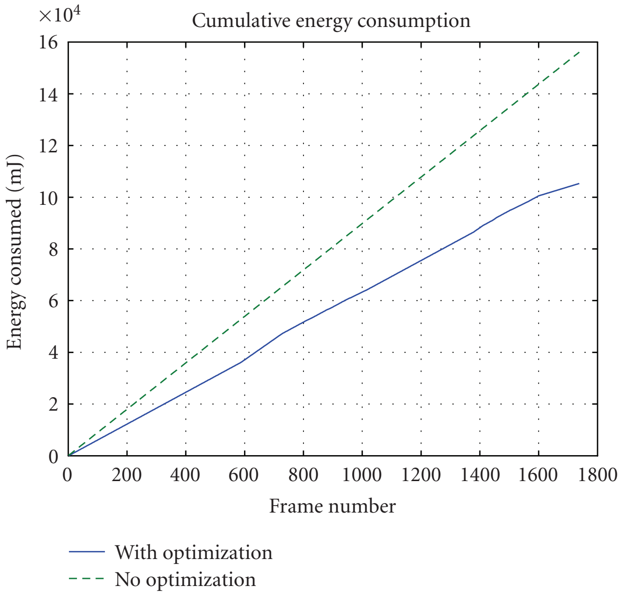


FIGURE 12: Energy consumption with and without optimization for *abc_news* video clip.

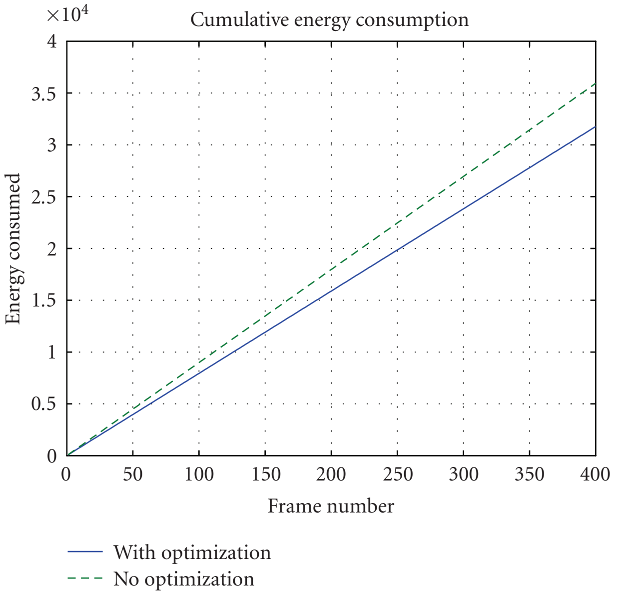


FIGURE 13: Energy consumption with and without optimization for *Foreman* video clip.

The proxy server includes four primary components—the video transcoder, the proposed QABS module, the signal multiplexer, 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 multiplexer 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.

On the mobile client, the demultiplexer 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. 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 a video sequence, based on his/her quality preference as well as concern for battery consumption.

Table 1: Results of QABS (G: good; F: fair; E: Excellent).

Alfa mean			Quality (dB)			Power saving(%)		
F	G	E	F	G	E	F	G	E
149	162	186	30.17	34.28	42.31	41.8%	36.7%	27.3%

4.2. *Measurement methodology*

For video quality and power measurements, we use the setup shown in Figure 9. The proxy in our experiments is a Linux desktop with a 1 GHz processor and 512 MB of RAM. All our measurements are made on a Compaq iPAQ 3650. In order to control the backlight and pixel luminance, we develop our own Linux-based API functions. We use a national instruments PCI DAQ board to sample voltage drops across a resistor and the iPAQ, and sample the voltage at 200 K samples/s. We calculate the instantaneous and average power consumption of the iPAQ using the formula $P_{\text{iPAQ}} = (V_R/R) \times V_{\text{iPAQ}}$.

4.3. *Experimental results*

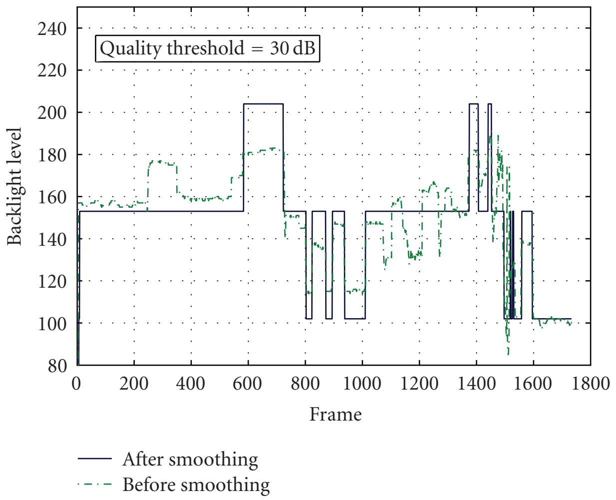
In our simulation, we use a video sequence captured from a broadcasted *ABC_news* program, whose first frame is shown in Figure 3(a). We chose this video as representative of a typical usage of a PDA—commuters watching the evening news on the way home. In Figure 10, we show the basic statistics (i.e., the mean and the variance of luminance per frame) of this video.

We assume that the users are given three quality options, fair, good, and excellent, which respectively correspond to the PSNR value of 30 dB, 35 dB, and 40 dB. After applying the algorithm "**Proc**: FastFindAlfa," we obtain the adapted Alfa* for these three quality preferences, as is shown in Figure 11. It can be seen that higher video quality needs higher backlight level on average.

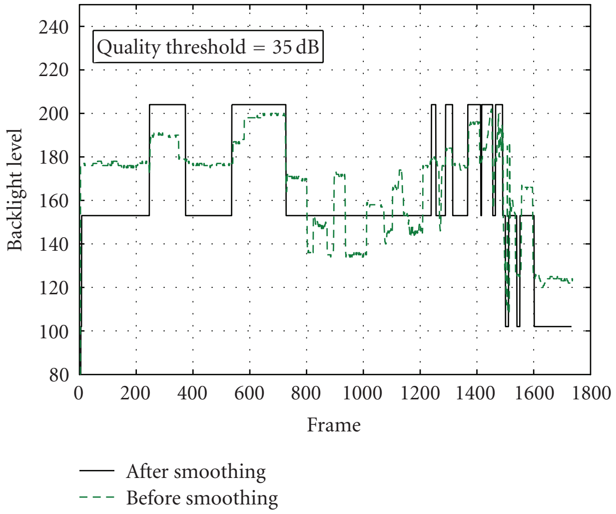
In Figure 14, we show Alfa* before and after the backlight smoothing process for different quality preferences. It is seen that the small variation and the abrupt change of the backlight switching are significantly eliminated after the filtering and quantization. In addition, as we expected, the backlight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

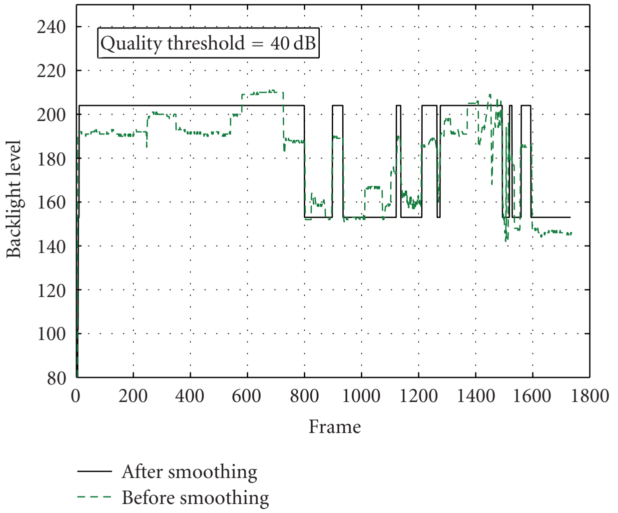
In Figure 15, we show that the filtering and quantization processes may lead to instantaneous quality fluctuation, which is contrasted to the consistent quality before backlight smoothing. Nevertheless, we observe that the quality fluctuation is around the designated quality threshold and mostly happens at scene changes.



(a) Alfa* for fair quality

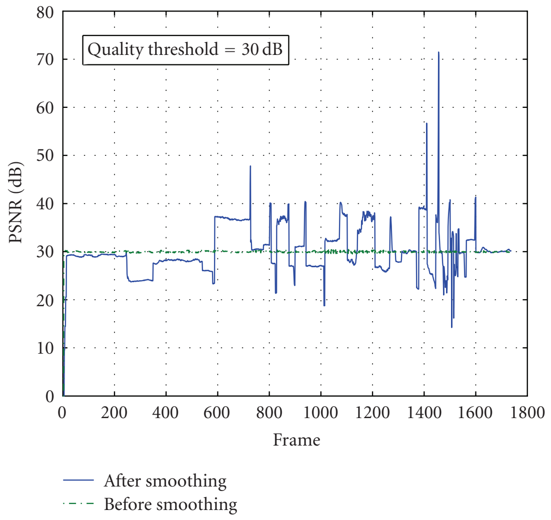


(b) Alfa* for good quality

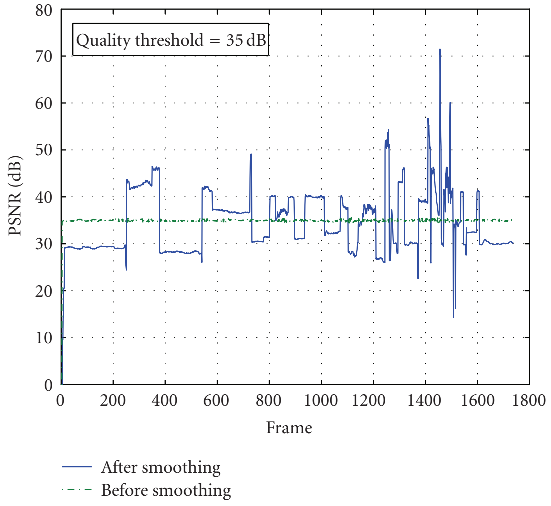


(c) Alfa* for excellent quality

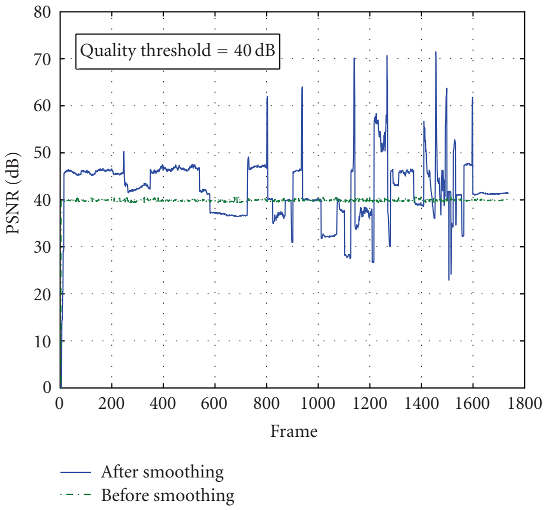
Figure 14: Optimized Backlight level before and after filtering and quantization.



(a) Fair quality



(b) Good quality



(c) Excellent quality

Figure 15: Actual PSNR quality before and after filtering and quantization.

In Figures 12 and 13, we compare the actual energy consumption on a Compaq ipaq with and without our quality aware backlight adaptation. As seen from the graph, the energy savings from our backlight adaptation for the ABCNews clip is 35%–40% of the total energy consumed due to backlight. Even for videos that offer very little opportunity to aggressively perform backlight adaptation (e.g., foreman video clip, which is simply a talking head), we can achieve energy savings as high as 14–20% with negligibal video quality sacrificing.

5. CONCLUSION

In this paper, we apply a backlight scaling technique to a proxy-based video streaming framework. We explicitly associate backlight switching to the perceptual video quality in 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 switching, we also propose two supplementary schemes to smooth the backlight switch process such that the user perception of the video stream can be substantially improved. Our experiment shows that by applying our scheme, up to 40% power can be saved with negligibal video quality sacrificing.

ACKNOWLEDGMENTS

The authors would like to thank Stefano Bossi and Michael Philpott who helped them with the simulations and experimental system setup. This work was partially supported by National Science Foundation.

REFERENCES

[1] A. Azevedo, R. Cornea, I. Issenin, et al., "Architectural and compiler strategies for dynamic power management in COPPER project," in *Proceedings of International Workshop on Innovative Architecture (IWIA '01)*, Maui, Hawaii, USA, January 2001.

[2] K. Choi, W.-C. Cheng, and M. Pedram, "Frame-based dynamic voltage and frequency scaling for an MPEG player," *Journal of Low Power Electronics*, vol. 1, no. 1, pp. 27–43, 2005.

[3] S. Chandra, "Wireless network interface energy consumption: implications for popular streaming formats," *Multimedia Systems*, vol. 9, no. 2, pp. 185–201, 2003.

[4] M. Stemm, P. Gauthier, D. Harada, and R. H. Katz, "Reducing power consumption of network interfaces in hand-held devices," in *Proceedings of the 3rd International Workshop on Mobile Multimedia Communications (MoMuc '96)*, Princeton, NJ, USA, September 1996.

[5] S. Pasricha, M. Luthra, S. Mohapatra, N. Dutt, and N. Venkatasubramanian, "Dynamic backlight adaptation for low-power handheld devices," *IEEE Design and Test of Computers*, vol. 21, no. 5, pp. 398–405, 2004.

[6] N. Chang, I. Choi, and H. Shim, "DLS: dynamic backlight luminance scaling of liquid crystal display," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 8, pp. 837–846, 2004.

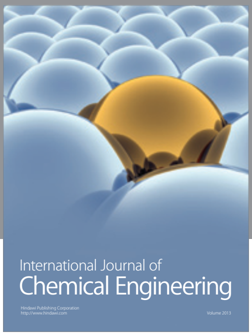
[7] W.-C. Cheng, Y. Hou, and M. Pedram, "Power minimization in a backlit TFT-LCD display by concurrent brightness and

contrast scaling," in *Proceedings of Design, Automation and Test in Europe Conference and Exhibition (DATE '04)*, vol. 1, pp. 252–257, Paris, France, February 2004.

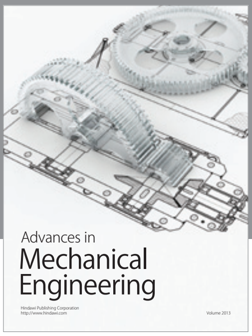
[8] S. Wolf and M. H. Pinson, "Spatial-temporal distortion metrics for in-service quality monitoring of any digital video system," in *Proceedings of the Multimedia Systems and Applications II*, vol. 3845 of *Proceedings of SPIE*, pp. 266–277, Boston, Mass, USA, September 1999.

[9] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on Broadcasting*, vol. 50, no. 3, pp. 312–322, 2004.

[10] J. L. Zachary, *Introduction to Scientific Programming: Computational Problem Solving Using Maple and C.*, Telos Publishers, Surrey, UK, 1996.

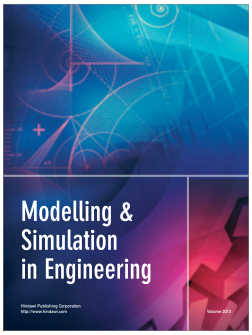
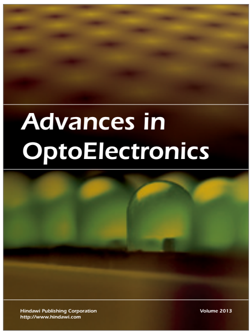
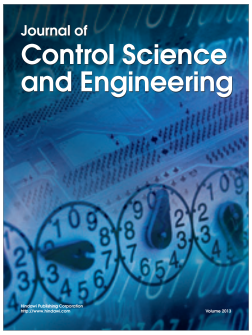

International Journal of
Distributed Sensor Networks

International Journal of
Chemical Engineering

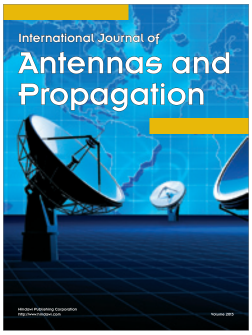
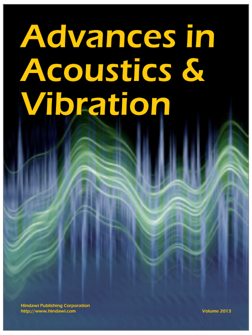
International Journal of
Rotating Machinery

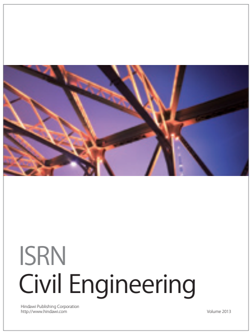
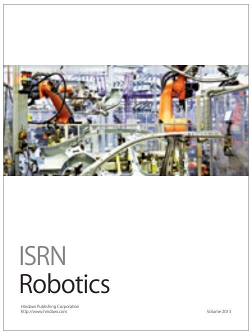
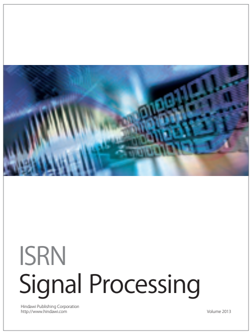
The Scientific World Journal

Advances in
Mechanical Engineering

VLSI Design

Journal of
Electrical and Computer Engineering


Active and Passive Electronic Components

Modelling & Simulation in Engineering

Advances in OptoElectronics

Journal of
Control Science and Engineering

Journal of
Sensors

International Journal of
Antennas and Propagation

Advances in Acoustics & Vibration

ISRN
Electronics

ISRN
Civil Engineering

ISRN
Robotics

ISRN
Signal Processing

ISRN
Sensor Networks