

# Dynamic Backlight Scaling Optimization for Mobile Streaming Applications

Pi-Cheng Hsiu, Chun-Han Lin, and Cheng-Kang Hsieh

Research Center for Information Technology Innovation, Academia Sinica, Taipei, Taiwan 115, R.O.C.

{pchsiu, chunhan, changun}@citi.sinica.edu.tw

**Abstract**—With the increasing variety of mobile applications, reducing the energy consumption of mobile devices is a major challenge in sustaining multimedia streaming applications. This paper explores backlight scaling, which is deemed a promising technical solution. First, we model the problem as a dynamic backlight scaling optimization problem. The objective is to minimize the energy consumption of the backlight when displaying a video stream without adversely impacting the user’s visual perception. Then, we propose a dynamic-programming algorithm to solve the fundamental problem and prove its optimality in terms of energy savings. Finally, based on the algorithm, we consider implementation issues. We have also developed a prototype implementation integrated with existing video streaming services to validate the practicability of the approach. The results of experiments conducted to demonstrate the efficacy of the proposed algorithm are very encouraging.

**Index Terms**—Energy-efficient optimization, dynamic backlight scaling, multimedia streaming applications, mobile devices

## I. INTRODUCTION

Advances in information and communications technology have increased the popularity of mobile devices. This in turn is motivating the development of a growing number of mobile applications and services, which are having a profound effect on people’s lifestyles. However, reducing the energy consumption of mobile devices that utilize the mobile applications is a major challenge. Recent studies on mobile user activity indicated that the backlight used to illuminate the display subsystem consumes most of the energy; thus, it should receive the most attention with respect to improving energy efficiency [19], [20]. Mobile users are becoming increasingly addicted to multimedia streaming applications, such as YouTube [6], and the ability to disseminate videos via social network communities like Facebook [3]. A recent report [12] forecast that mobile data traffic will double every year in the coming few years, and video streaming will account for almost 66 percent of the traffic by 2014. Such usage behavior will lead to a significant increase in the energy consumption of mobile devices, especially with the strong demand for larger, higher-resolution screens. This observation motivates us to study *dynamic backlight scaling* for multimedia streaming applications on mobile devices.

The display subsystem needs to stay in active mode for as long as the video stream is displayed; thus, a sensible way to reduce the energy consumption is to dim the backlight. However, this may lead to image distortion, which is normally defined as the resemblance between the original video image and the backlight-scaled image [9], [21]. For example, the *structural similarity index* [22], a metric specially designed to comply with the perception of the human eye, is widely used to assess distortion. In recent years, a number of effective *backlight scaling techniques* have been developed to limit the distortion and/or maintain the fidelity of a single image when the backlight is dimmed. In particular, the *just noticeable difference*<sup>1</sup>

of the *human visual system* is exploited by the approach in [16], so that the incurred image distortion is confined to a tolerable threshold and does not affect the clarity of the display significantly. Various *image compensation techniques*, e.g., [10], [11], [13], have also been proposed to further dim the backlight. They compensate for the image distortion through image pixel transformation and further reduce the energy consumption at the same time. These techniques determine the dimmest backlight level for a single image and provide a foundation for exploring dynamic backlight scaling optimization in this paper.

A video stream comprises a series of image frames. An intuitive way to reduce energy consumption is to treat a video stream as a collection of images and dynamically change the backlight by applying backlight scaling techniques to each image frame individually [7], [18]. However, in most video applications, the dimmest backlight level may vary significantly across consecutive frames [11], so changing the backlight dynamically over a number of frames may result in flickering effects [15]. To resolve the issue, some approaches determine the backlight level for an image frame by considering the preceding frame’s pixel values and backlight level [14], [15]. The drawback of this strategy is that switching the backlight level frequently may introduce inter-frame brightness distortion [9]. Furthermore, the hardware and software require some time to react and adjust the backlight, so it is necessary to reduce the frequency of backlight switching [17]. To this end, the approach in [17] groups the image frames of a video and determines a common backlight level for each group. As a result, the backlight of a scene may be changed suddenly if the frames comprising the scene are partitioned into different groups. In contrast, the approach in [9] quantizes the number of backlight levels to eliminate small backlight fluctuations during a scene and, thereby prevents frequent backlight changes. The drawbacks of existing heuristics result primarily from determining the backlight level of each image frame based on its adjacent frames (and itself), instead of having an overall consideration based on all the frames in a video. Approaches based on heuristic or empirical studies cannot provide a rigid theoretic framework for dynamic backlight scaling optimization.

In this paper, we focus on minimizing the energy consumption incurred by the backlight for multimedia streaming applications on mobile devices, without adversely impacting the user’s visual perception. The contributions of this study are as follows. First, we model the problem of dynamic backlight scaling optimization that imposes two scaling constraints on the backlight changes over image frames. Second, we propose an algorithm based on dynamic programming to solve the fundamental problem. The solution involves determining the appropriate backlight levels for image frames without violating the scaling constraints. We prove that the algorithm is optimal in terms of energy savings when the energy consumption is a monotonic function of the backlight levels. Third, we discuss technical implementation issues that arise from incorporating the proposed algorithm into existing video streaming services, such as YouTube. We have also

<sup>1</sup>This represents the minimum amount by which the stimulus intensity must be changed in order to produce a noticeable variation in sensory experience.

developed a prototype implementation (comprised of an on-line backlight server and a mobile application program) based on Android smartphones [1] to demonstrate the efficacy of the proposed algorithm and validate the practicability of the approach studied in this work. When the program is installed, an Android smartphone of HTC Desire [4] can achieve a significant energy reduction when browsing video streams on YouTube [6], but users are not aware that dynamic backlight scaling is being applied. Finally, we conducted a series of experiments and compared the proposed algorithm with a heuristic revised based on the approach proposed in [17]. The experimental results provide further insights into dynamic backlight scaling on mobile devices for multimedia streaming applications.

The remainder of this paper is organized as follows: Section II describes the system model and defines the problem. In Section III, we propose an optimal algorithm to solve the problem. In Section IV, we discuss some system implementations. The experimental results are reported in Section V. Section VI contains some concluding remarks.

## II. SYSTEM MODEL AND PROBLEM DEFINITION

In this section, we present the respective scaling constraints that limit the dimmest backlight level of each image frame and the minimum time period (duration) between two adjacent backlight changes. We also introduce the power model and define the fundamental problem.

A streaming video comprises a series of image frames displayed in succession at a constant rate. Each image frame is represented by a grid of pixels. The *perceptual luminance intensity* of a pixel shown on a display panel is proportional to the product of the *backlight level* and the *pixel luminance*<sup>2</sup> [11], [17]. The pixel luminance does not have a noticeable impact on the energy consumption, but the backlight level is a decisive factor. Therefore, dimming the backlight level while compensating for the loss of the perceptual luminance intensity by increasing the pixel luminance is considered an effective way to save energy in mobile devices. A number of techniques have been proposed to limit image distortion and/or maintain image fidelity, e.g., [10], [11], [13], [16]. In this paper, we simply assume that each image frame is associated with a *critical backlight level* that represents its dimmest backlight level (determined by the above techniques), and treat the critical backlight level as a scaling constraint.

A display subsystem relies on a light source to illuminate the display from behind. Although the light source is usually capable of adjusting the luminance levels, it may not be able to adjust the backlight for every frame because it takes some time to react. In addition, frequent switching of the backlight may introduce inter-frame brightness distortion [9]. Thus, there needs to be a minimum duration between consecutive changes in backlight levels. We define another scaling constraint that requires the adjusted backlight level resulting from each backlight change to remain the same for a certain number of subsequent image frames. For ease of presentation, we map the backlight levels into corresponding dimming values and normalize them in the range 0 to 1, with 0 representing no backlight and 1 representing the full backlight.

The energy consumed by a display subsystem is dominated by the backlight level. Some studies simply assume that the energy consumption is linearly proportional to the backlight level and model the relation with liner functions [8], [9]. However, as observed in [11], although the energy consumption increases with the backlight level,

<sup>2</sup>A pixel's luminance is an 8-bit value between 0 and 255. It can be derived by converting its RGB values to the  $YCbCr$  coordinate space with standard conversion functions [17].

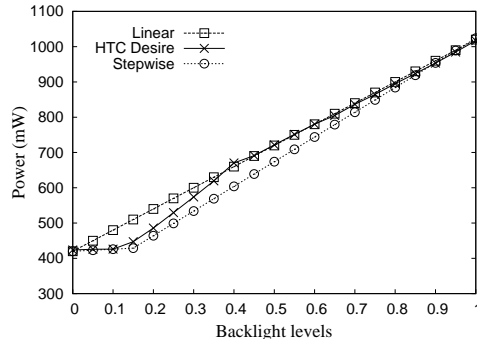


Fig. 1. Power models

the increasing slopes may change within different ranges of backlight levels. Various stepwise functions have been used to characterize the power models of display subsystems [13], [15], [16]. Figure 1 shows the power model measured from an Android smartphone of HTC Desire [4] in practice, as well as the models approximated by linear and stepwise functions. In this paper, we make no assumption about the power model to which the proposed algorithm can be applied. However, we prove that the algorithm is optimal in terms of energy saving if the energy consumption increases monotonically with the backlight level.

Scaling down the backlight will save energy; however, it could affect the user's visual experience if appropriate backlight levels are not applied to the image frames. Thus, our objective is to determine an appropriate backlight level for each image frame such that the total energy consumption of the backlight is minimized. A mapping of a given set of image frames to the available backlight levels (in the range 0 to 1) is called an *assignment*. An assignment is *feasible* if two conditions are satisfied: (1) the backlight level applied to any image frame is no lower than the frame's critical backlight level; and (2) the number of image frames between any two backlight changes is not less than a specified number. Next, we formally define the problem:

### The Dynamic Backlight Scaling Optimization Problem

*Instance:* A set of  $N$  image frames  $F = \{f_1, f_2, \dots, f_N\}$ , where each frame  $f_i$  is associated with a critical backlight level  $c(i)$ ; a minimum duration of  $d$  frames for a backlight change; and a power model  $P()$ , which represents the relation between the backlight levels and the energy consumption.

*Objective:* A feasible assignment  $\sigma$  such that the total energy consumption  $\sum_{i=1}^N P(\sigma(i))$  is minimized.

## III. DYNAMIC BACKLIGHT SCALING OPTIMIZATION

### A. Algorithm Description

In this section, we present a dynamic-programming algorithm and its polynomial-time implementation to solve the dynamic backlight scaling optimization problem. The algorithm is based on the recursive formula given in Equation (1). Let  $E(i)$  be the minimum energy required to display the first  $i$  image frames without violating the scaling constraints, provided that a backlight change is allowed at frame  $f_{i+1}$ . For ease of presentation, we introduce *blank frames* and append them to the given image frame set. The critical backlight level of a blank frame is set at 0, i.e.,  $c(i) = 0, \forall i > N$ . It is assumed that the blank frames do not consume any energy regardless of the backlight levels applied to them. We delineate two possible cases of Equation (1):

$$E(i) = \begin{cases} i \times P(\max_{\forall 1 \leq k \leq i} c(k)), & \text{if } 1 \leq i \leq d; \\ \min_{\forall 1 \leq j \leq i-d} \{E(j) + (\min(i, N) - \min(j, N)) \times P(\max_{\forall j < k \leq i} c(k))\}, & \text{otherwise.} \end{cases} \quad (1)$$

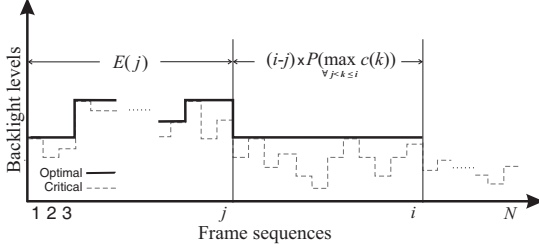


Fig. 2. An illustration of the dynamic-programming formula

- (1) If  $1 \leq i \leq d$ , then  $E(i)$  is set at  $i \times P(\max_{\forall 1 \leq k \leq i} c(k))$ . That is, the  $i$  frames are provided with a common backlight level, which is equal to the maximum of their critical backlight levels.
- (2) Otherwise, if  $i > d$ , suppose that the last  $i - j$  frames are assigned the same backlight level. Because any frame  $f_k$ ,  $\forall k > N$ , is a blank frame that does not consume any energy, we should only consider  $\min(i, N) - \min(j, N)$  frames when computing the energy consumption of the  $i - j$  frames. The minimum energy consumption of the first  $j$  frames is  $E(j)$ , provided that a backlight change is allowed at frame  $f_{j+1}$ . Thus, the total energy consumption of the  $i$  frames is  $E(j) + (\min(i, N) - \min(j, N)) \times P(\max_{\forall j < k \leq i} c(k))$ , as shown by the example in Figure 2. Note that after the backlight level is changed, it must remain at that level for at least  $d$  frames. To ensure that the backlight can be changed at frame  $f_{i+1}$ , the possible  $j$  values must be in the range 1 to  $i - d$ . By considering all of them,  $E(i)$  is set as the minimum energy consumption.

The objective is to derive  $E(N + d - 1)$ . Note that the  $d - 1$  blank frames consume no energy, and are introduced to relax the presupposition that the last  $d$  frames of the  $N$  frames have the same backlight level. Consequently,  $E(N + d - 1)$  is equal to the minimum energy required to display the original  $N$  image frames without violating the scaling constraints.

Algorithm 1 implements the dynamic-programming formula recursively. Once derived, the solution to each subproblem  $E(i)$  is stored in a corresponding entry in a one-dimensional table  $S$ . At the beginning of the algorithm, each table entry is initialized as  $\infty$  to indicate that the corresponding subproblem has not been solved yet. Whenever Procedure  $E(i)$  is invoked, the procedure simply returns the previously derived solution stored in table entry  $S[i]$  if the entry has been updated. Otherwise, the solution to the subproblem is derived based on the presented dynamic-programming formula and returned. After the entire table has been derived, a corresponding feasible assignment  $\sigma$  can be constructed by back tracing the table  $S$  based on the dynamic-programming formula as follows. We begin with the last table entry by setting an index  $i$  as  $N + d - 1$ , and examine each entry sequentially (from  $S[i - d]$  to  $S[1]$ ) until we find an entry  $S[j]$  in which the stored value minimizes the solution to subproblem  $E(i)$ . The  $i - j$  frames,  $f_{j+1}, f_{j+2}, \dots, f_i$ , are assigned the same backlight level  $\max_{\forall j < k \leq i} c(k)$ . We then start with the discovered entry by setting  $i$  as  $j$  and repeat the above process recursively until  $i \leq d$ . Finally, we assign the remaining  $i$

frames,  $f_1, f_2, \dots, f_i$ , the same backlight level  $\max_{\forall 1 \leq k \leq i} c(k)$ . Because we have to examine  $N$  table entries at most, and each examination takes constant time  $O(1)$  (refer to the following time complexity analysis), the construction of a feasible assignment  $\sigma$  based on  $S$  can be completed in  $O(N)$  time.

#### Algorithm 1:

**Input:** A frame set  $F$  with an association function  $c()$ , a minimum duration  $d$ , and a power model  $P()$

**Output:** The minimum energy consumption for a feasible assignment  $\sigma$

```

for  $i \leftarrow 1$  to  $N + d - 1$  do
   $S[i] \leftarrow \infty$ 
end for
return  $E(N + d - 1)$ 

```

#### Procedure $E(i)$

```

if  $S[i] < \infty$  then
  return  $S[i]$ 
end if
if  $1 \leq i \leq d$  then
   $S[i] \leftarrow i \times P(\max_{\forall 1 \leq k \leq i} c(k))$ 
else
   $S[i] \leftarrow \min_{\forall 1 \leq j \leq i-d} \{E(j) + (\min(i, N) - \min(j, N)) \times P(\max_{\forall j < k \leq i} c(k))\}$ 
end if
return  $S[i]$ 

```

#### B. Properties

In the remainder of this section, we analyze the time complexity of Algorithm 1 and prove its optimality for the dynamic backlight scaling problem.

*Lemma 1:* The time complexity of Algorithm 1 is  $O(N^2)$ .

**Proof.** It is assumed that  $d < N$ ; otherwise, the trivial case is solvable in  $O(N)$  time. The time complexity of the implementation depends on the number of table entries and the time required to derive the solution to a subproblem. The table contains  $O(N)$  entries, each of which is initialized and stored with the solution to the corresponding subproblem once. The solution to each subproblem  $E(i)$  is derived by referring to  $i - d$  (which is less than  $N$ ) preceding entries sequentially. When an entry  $S[j]$  is referred to, computing a possible solution to  $E(i)$  takes  $O(1)$  time. Note that  $\max_{\forall j < k \leq i} c(k)$  can be determined in constant time by simply comparing  $c(j + 1)$  and  $\max_{\forall j+1 < k \leq i} c(k)$ , because the latter term has been determined when entry  $S[j + 1]$  was referred to. Thus, deriving the solution to any subproblem takes  $O(N)$  time. In summary, table  $S$  can be constructed in  $O(N^2)$  time. ■

*Theorem 1:* Algorithm 1 is an optimum algorithm for the dynamic backlight scaling problem.

**Proof.** The theorem follows directly from the correctness of the dynamic-programming formula  $E(i)$ . We prove the correctness of the formula by mathematical induction on the index  $i$ . As the induction basis,  $1 \leq i \leq d$ , to ensure that a backlight change is allowed at frame  $f_{i+1}$ , the first  $i$  frames must be assigned the same level. Since  $P()$  is a monotonic function of the backlight levels, the minimum energy

consumption required to display the  $i$  frames is  $i \times P(\max_{1 \leq k \leq i} c(k))$ . Thus, the formula is correct. For the induction hypothesis, suppose that the formula is always correct for the first  $i$  frames when  $i < n$ . We show that the formula is also correct for the first  $n$  frames.

Suppose that the last  $n-j$  frames are assigned a common backlight level. The minimum energy consumption required to display the  $n-j$  frames is  $(\min(n, N) - \min(j, N)) \times P(\max_{j < k \leq n} c(k))$ . Since  $j < n$ , by the induction hypothesis,  $E(j)$  is the minimum energy consumption for the first  $j$  frames, provided that a backlight change is allowed at frame  $f_{j+1}$ . Since the common level applied to the last  $n-j$  frames is not subject to the backlight level assigned to frame  $j$ , the total energy consumption of the  $n$  frames is the sum of the energy consumption of the first  $j$  frames and that of the last  $n-j$  frames. To allow a backlight change at frame  $f_{n+1}$ , we must ensure that  $n-j \geq d$ . This implies that all possible  $j$  values are in the range 1 to  $n-d$ . By considering all of them,  $E(n)$  is set as the minimum energy consumption yielded. Hence,  $E(n)$  is the minimum energy consumption of the first  $n$  frames, provided that a backlight change is allowed at frame  $f_{n+1}$ . The theorem follows. ■

#### IV. IMPLEMENTATION ISSUES

In this section, we discuss some implementation issues that arise when integrating the proposed algorithm with existing video streaming services. Determining the backlight assignment for a video is computation-intensive; in particular, computing critical backlight levels usually involves analyzing a large number of image pixels. Thus, the energy consumption incurred by the computation could easily negate the energy savings gained by the dynamic backlight scaling technique if the assignment is computed on mobile devices. Consequentially, the primary principle is that a designated server is responsible for computing the critical backlight levels and running the proposed algorithm. We delineate two possible application scenarios. First, if modification of the video content is allowed, the server determines the backlight assignment for the video and embeds the information in the corresponding headers of the image frames in advance. Then, when the video is displayed on a mobile device, the video player in the device simply adjusts the backlight level according to the embedded information. In the second scenario, the video content cannot be modified, so the server stores the backlight information in a corresponding text file (called the *backlight file*) instead. When a video is to be displayed on a mobile device, the corresponding backlight file is transmitted with the video. The concept is similar to that of subtitles in a video. In the first scenario, the video streaming provider, such as YouTube [6], could determine the backlight assignments for the videos, while the second scenario could be developed as a value-added service of the Internet service provider like AT&T [2].

We have implemented a prototype system based on the second scenario<sup>3</sup>. The prototype system includes an on-line backlight server and a mobile application program based on the Android platform [1]. The backlight server automatically uses the proposed algorithm to analyze the videos on major video streaming websites. Each derived backlight assignment is stored in a space-efficient format and associated with the corresponding video's URL link. Because the size of a backlight file for a 15MB video (with a bit rate of 550-650kbps) is usually less than 1 KB, it can be transmitted in no time when

<sup>3</sup>The energy-saving service is not intended for all models of mobile devices, but the popular ones and those needing the service. However, the accuracy of the power model will only affect the amount of energy saved, not the visual perception, so other device models could also benefit from the technique.

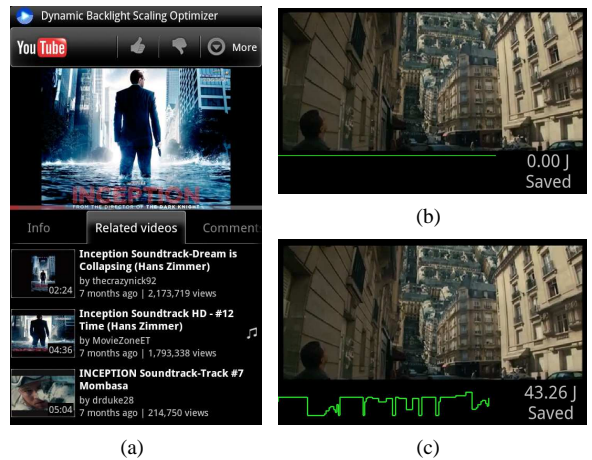


Fig. 3. Snapshots of the mobile application program

the wireless bandwidth is sufficient for video streaming. It would be ideal if all the videos on streaming websites could be analyzed in advance, but doing so would be tremendously timing-consuming and a huge storage space would be required. Hence, we analyzed the most popular videos on YouTube, and left the remainder to be analyzed on demand. To support on-demand video analysis, we have developed an application program that runs on Android mobile devices, as shown in Figure 3(a). When a user starts to play a video stream on a smartphone with the program installed, the program sends the video's URL link to the backlight server. If there is no corresponding backlight file, the server replies accordingly and starts analyzing the video to generate the corresponding backlight file. In this case, the video is played without dynamic backlight scaling, as shown in Figure 3(b). Conversely, if the server returns the corresponding backlight file, the program adjusts the backlight dynamically according to the backlight file with the video being played on the smartphone, as shown in Figure 3(c).

#### V. PERFORMANCE EVALUATION

##### A. Experimental Setup

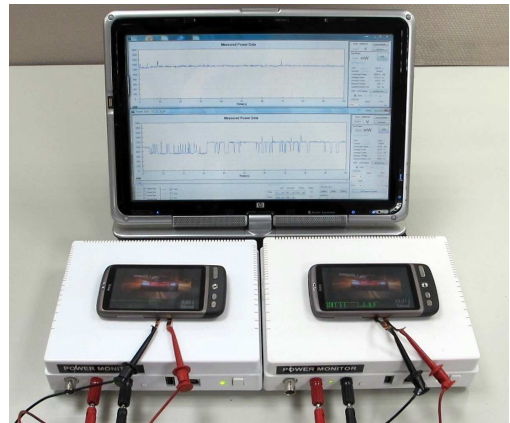


Fig. 4. The experimental environment

To better understand the properties of, and gain insights into, dynamic backlight scaling for mobile streaming applications, we evaluated the performance of the proposed algorithm via several case studies. Any image compensation technique<sup>4</sup> could be used to

<sup>4</sup>When an image compensation technique is used, the video player's decoder must be modified to increase the pixel luminance when playing videos.

compute a video’s critical backlight levels. However, to exclude the energy reduction benefits from the adopted compensation technique, a simple image distortion metric should be employed. In the case studies, we adopted the structural similarity (SSIM) index [22] as the image distortion metric because of its popularity in related studies [7], [18]. The resultant SSIM index is a decimal value between -1 and 1, where the value 1 is only reachable in the case of two identical sets of data. Given a video stream, the critical backlight level of each image frame was computed with respect to a specified SSIM index; then, the proposed algorithm was used to determine the optimal assignment with respect to the critical backlight levels. The mobile application program was installed on an Android smartphone of HTC Desire [4], equipped with a 3.7-inch Super LCD display subsystem. The power model with 20 backlight levels is shown in Figure 1. In the experimental environment, shown in Figure 4, one smartphone performed dynamic backlight scaling while the other did not, and the energy consumption of their display subsystems was measured by Power Monitors of Monsoon Solutions [5].

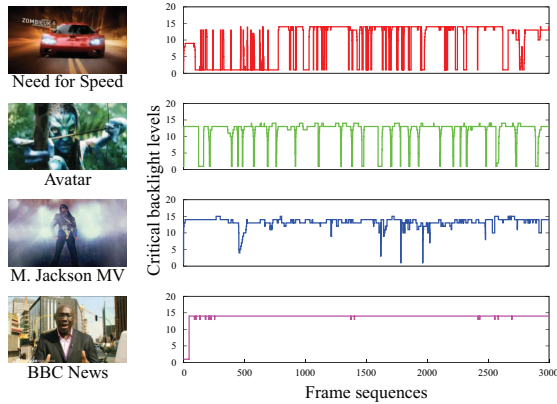


Fig. 5. Snapshots and critical backlight levels of four videos

We studied four videos with representative characteristics, namely, Need for Speed, Avatar, M. Jackson MV, and BBC News, all of which were found on YouTube. Figure 5 shows some snapshots of the videos and the critical backlight levels when the SSIM index was set at 0.9. Need for Speed, which is a typical advertisement for video games, comprises diverse scenes that demonstrate the game’s high fidelity; thus, the critical backlight levels vary significantly. Avatar is a characteristic trailer for sci-fi movies, so the high fidelity feature is also emphasized, but the consecutive scenes are longer and more similar than those of game advertisements. M. Jackson MV is a classic performance of Michael Jackson with fantastic stage effects. The critical backlight levels vary frequently, but the changes are small. BBC News is a news video clip comprised of mainly static scenes. The performance metric was the percentage of energy savings achieved without adversely impacting the visual perception. We investigated the impacts of the SSIM index within the range 0.8 to 0.98 when the minimum duration,  $d$ , was set at 10 frames. The impact of setting the minimum duration between 4 and 90 frames was also investigated when the SSIM index was set at 0.9. The settings were based on the following observations. HTC Desire allows backlight changes up to 8 times per second, which corresponds to a minimum duration of 4 frames for 30 frame-per-second videos. Moreover, it was difficult to determine whether a video adopted our technique when the SSIM index was no less than 0.8; and it was difficult to differentiate which of two videos adopted our technique when the

SSIM index was no less than 0.9.

In addition to the case studies, we compared the proposed algorithm, denoted as *OPT*, with a heuristic-based algorithm, denoted as *GOS*, when the SSIM index was set at 0.9 and the minimum duration was set at 10, 30, and 60. *GOS* is based on the concept of groups of scenes utilized by the approach in [17]. It starts with a group of  $d$  frames, and then adds subsequent frames until the variance of the average luminosity exceeds a threshold. The process is repeated until all frames have been grouped, and the frames in each group are assigned the maximum of their critical backlight levels. Following [17], the variation threshold was set at 40. We adopted *GOS* for comparison because the minimum duration between changes in backlight levels can be ensured by setting the minimum group size.

### B. Experimental Results

Figure 6 shows the impacts of the SSIM index on the energy savings achieved by *OPT*. As expected, the energy savings increased as the SSIM index decreased. The reason was that a smaller SSIM index led to lower critical backlight levels. This in turn implied more energy savings, because a feasible assignment with respect to higher critical backlight levels was also feasible to lower critical backlight levels, but not vice versa. The results show that, when  $SSIM = 0.9$ , *OPT* can achieve energy savings between 18% and 31%, depending on the characteristics of videos. The percentage of energy savings was more evident when a video’s critical backlight levels varied significantly, such as in the Need for Speed video. This was because the significant variation meant a large number of low critical backlight levels, which provided opportunities to dim the backlight. Interestingly, a video with a large number of static scenes, such as BBC News, also benefited substantially from the dynamic backlight scaling technique due to the difference between the full and critical backlight levels.

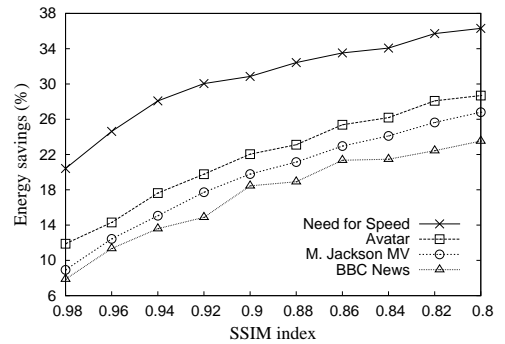


Fig. 6. Impacts of the SSIM index

Figure 7 shows the impacts of the minimum duration on the energy savings achieved by *OPT*. The energy savings generally decreased as the minimum duration increased. This is because the solution space under a larger duration is a subset of that under a smaller duration. We observed an interesting exception with the M. Jackson MV when  $d = 4$ . The phenomenon occurred because the video’s critical backlight levels varied frequently, but the changes were small. The switching overhead, although almost negligible, could negate the energy gain if changing the backlight frequently only yielded marginal energy savings. We also observed that the minimum duration affected the energy savings of the Need for Speed and Avatar videos significantly; however, it did not have a significant impact on M. Jackson MV and BBC News. The results imply that the first two videos contained a large variety of scenes, each comprised of a few frames. In contrast,

the other two videos contained longer scenes with similar critical backlight levels. In addition, as in the previous experiment, the more diverse the critical backlight levels, the greater the energy savings. The results show that, when  $d = 10$ , OPT can achieve energy savings between 18% and 31%. This is similar to the result in Figure 6 when SSIM = 0.9 because the same setting was used in both experiments.

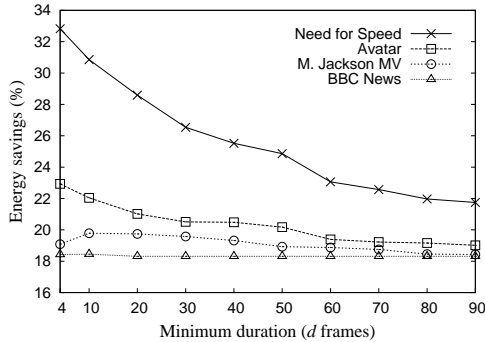


Fig. 7. Impacts of the minimum duration

Figure 8 shows the energy savings achieved by OPT and GOS for videos with different characteristics. As expected, OPT outperformed GOS in all cases because it has proved optimal in terms of energy savings. For the same video, the performance difference between the two algorithms was generally more evident when  $d$  was small, since the number of possible solutions increased as the minimum duration decreased. Consequentially, it was harder for the heuristic algorithm, GOS, to make a “good” decision. This also explains why, for the same minimum duration, the performance difference was more evident when a video’s critical backlight levels varied significantly. The results show that, for Need for Speed video, OPT reduced the energy consumption 1.22, 1.19, and 1.15 times more than GOS when  $d = 10, 30$ , and  $60$ , respectively. We also observed that, under GOS, a scene may suffer from sudden backlight changes if it covered two groups and the maximum critical backlight levels of the groups were very different.

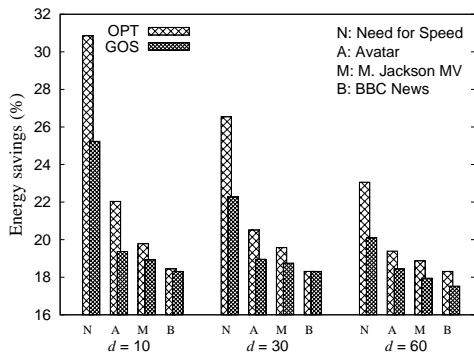


Fig. 8. Energy savings achieved by OPT and GOS for different videos

## VI. CONCLUDING REMARKS

This paper proposes an approach that minimizes the energy consumption incurred by the backlight when users access multimedia streaming on mobile devices. Specifically, the approach exploits backlight scaling and models the energy consumption as a fundamental optimization problem with scaling constraints (to limit image distortion and reflect hardware/software limitations). To solve the problem, we propose a dynamic-programming algorithm, and

prove that it is optimal in terms of energy savings when the energy consumption increases monotonically with the backlight levels. To validate the practicability of the approach, based on the algorithm, we implemented a mobile application program and deployed a backlight server that can be integrated with existing video streaming services. With the program installed, an HTC Desire mobile phone [4] achieved energy savings of 18-31% when browsing videos on YouTube [6], while users were not conscious of the dynamic backlight scaling technique. The efficacy of the proposed algorithm is more evident when a video contains a large variety of scenes. In addition to the above results, a series of experiments provided useful insights into dynamic backlight scaling for multimedia streaming applications on mobile devices.

In the future, we will release the mobile application program in the Android Market [1] and seek feedback on its performance and to identify more issues in this research direction.

## REFERENCES

- [1] Android. <http://www.android.com>.
- [2] AT&T Inc. <http://www.att.com/>.
- [3] Facebook. <http://www.facebook.com>.
- [4] HTC Corporation. <http://www.htc.com>.
- [5] Monsoon Solutions, Inc. <http://www.msoon.com>.
- [6] YouTube. <http://www.youtube.com>.
- [7] A. Bartolini, M. Ruggiero, and L. Benini. HVS-DBS: Human Visual System-Aware Dynamic Luminance Backlight Scaling for Video Streaming Applications. In *Proc. of the ACM EMSOFT*, pages 21–28, 2009.
- [8] N. Chang, I. Choi, and H. Shim. DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display. *IEEE Trans. on VLSI Systems*, 12(8):837–846, 2004.
- [9] L. Cheng, S. Mohapatra, M. E. Zarki, N. Dutt, and N. Venkatasubramanian. Quality-Based Backlight Optimization for Video Playback on Handheld Devices. *Advances in Multimedia*, 2007.
- [10] W.-C. Cheng and C.-F. Chao. Minimization for LED-backlit TFT-LCDs. In *Proc. of the ACM/IEEE DAC*, pages 608–611, 2006.
- [11] W.-C. Cheng and M. Pedram. Power Minimization in a Backlit TFT-LCD Display by Concurrent Brightness and Contrast Scaling. *IEEE Trans. on Consumer Electronics*, 50(1):25–32, 2004.
- [12] Cisco Systems, Inc. Cisco Visual Networking Index: Forecast and Methodology 2009-2014. Technical report, 2010. <http://www.cisco.com>.
- [13] A. Iranli, H. Fatemi, and M. Pedram. HEBS: Histogram Equalization for Backlight Scaling. In *Proc. of the ACM/IEEE DATE*, pages 346–351, 2005.
- [14] A. Iranli, W. Lee, and M. Pedram. Backlight Dimming in Power-Aware Mobile Displays. In *Proc. of the ACM/IEEE DAC*, pages 604–607, 2006.
- [15] A. Iranli, W. Lee, and M. Pedram. HVS-Aware Dynamic Backlight Scaling in TFT-LCDs. *IEEE Trans. on VLSI Systems*, 14(10):1103–1116, 2006.
- [16] A. Iranli and M. Pedram. DTM: Dynamic Tone Mapping for Backlight Scaling. In *Proc. of ACM/IEEE DAC*, pages 612–616, 2005.
- [17] S. Pasricha, M. Luthra, S. Mohapatra, N. Dutt, and N. Venkatasubramanian. Dynamic Backlight Adaptation for Low-Power Handheld Devices. *IEEE Design & Test of Computers*, 21(5):398–405, 2004.
- [18] M. Ruggiero, A. Bartolini, and L. Benini. DBS4video: Dynamic Luminance Backlight Scaling based on Multi-Histogram Frame Characterization for Video Streaming Application. In *Proc. of the ACM EMSOFT*, pages 109–118, 2008.
- [19] A. Shye, B. Scholbrock, and G. Memik. Into the Wild: Studying Real User Activity Patterns to Guide Power Optimizations for Mobile Architectures. In *Proc. of the ACM/IEEE MICRO*, pages 168–178, 2009.
- [20] A. Shye, B. Scholbrock, G. Memik, and P. A. Dinda. Characterizing and Modeling User Activity on Smartphones: Summary. In *Proc. of the ACM SIGMETRICS*, pages 375–376, 2010.
- [21] P.-S. Tsai, C.-K. Liang, T.-H. Huang, and H. H. Chen. Image Enhancement for Backlight-Scaled TFT-LCD Displays. *IEEE Trans. on Circuits and Systems for Video Technology*, 19(4):574–583, 2009.
- [22] W. Zhou, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. on Image Processing*, 13(4):600–612, 2004.