International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 07 | July-2016 www.irjet.net p-ISSN: 2395-0072

Cloud Based Android Video Player for Dynamic Backlight Optimization

Ms. Bhavika Lahoti[1], Prof. Hanumant Pawar, MITCOE, PUNE[2]

[1] MS Bhavika Lahoti, dept. of Computer Engineering, MITCOE, Pune, India
[2]Prof. Hanumant Pawar, dept. of Computer Engineering MITCOE, Pune, India

---***---

Abstract - *—In these few decades smart phone usage has increased greatly. As usage has increased people are trying to overcome the limitations of these phones. Battery consumption is one of the limitations of smart phone. This paper exactly works on the same issue. Paper explains basic concept needs to study before scaling down backlight, using critical backlight level scales down the backlight so that energy saving is maximum. Finally service of optimization is deployed on cloud server. An android video player application is been developed to connect user with the cloud server. Paper gives the comparison of different algorithm for optimization and has used efficient algorithm among those. Result shows nearly 20-30 % of energy saving*

***Key Words*: Backlight optimization, Battery life, Cloud Service, Distortion, User perception, Battery aware devices, Smart phone, battery Consumption, backlight level, Android video player, Energy saving**

1. INTRODUCTION

Few decades ago mobile phones were one of the greatest inventions of human beings. They were used for easy remote communication. But as time had passed many inventions in mobile phone were done, from basic communicating medium, they are now used to solve basic life issues. This is the reason mobile phones are now known as smart phones. These smart phones help humans in every situation, whether it is contacting any person, finding any places, searching for something or even for entertainment. Smart phones are saviour.

As they have so many advantages there must be some limitations to these phones for example they need network to communicate, they need internet to find places, they need battery to keep on working etc. Battery consumption is the biggest problem while using smart phone, if its battery is dead its of no use. Also as smart phones screen size increases battery consumption increases and now-a-days everyone wants big screen phone that to with high resolution. So battery problem is obviously more in such phones.

Solution to this problem is to have extended batteries or to have power bank but these need to carry along with phone which sometimes get inconvenient. Rather than having external solution we can have something which will optimize battery consumption without carrying it externally. Some survey about consumption of battery has concluded that consumption of battery for different applications is different. If we have working device it will consume 16 % for CPU, 24 % for network, 28 % for display and 34 % for backlight. These results show that back-light consumes maximum precent of battery. Backlight is nothing but the display light and that is why we deemed it down when we are running out of battery. This is the first reason paper has backlight optimization algorithm. We have different applications installed for different purposes among those some are in built applications which like music player, camera, video player, gallery, sound recorder, etc. Battery drainage is maximum while playing video on the phone because it needs to have sound adjustment, backlight consumption and different frames adjustments. Also, usage of streaming applications has increased widely as videos are better to express anything. This is the second reason for this paper to work on video streaming application. Optimization of backlight is not a simple task because it can cause distortion in video frame, flickering effect and also has adverse effect on user perception, so before scaling down the backlight we should consider above factor so that user can not catch up difference between original video and optimized one. Video's distortion generally defined as resemblance between the original video and the optimized one. Similarly, flickering effect is that when backlight level changes continuously per frame which may lead backlight change from dimmest level to its maximum level which will cause bad effects on users eye sight.

1.1 Outline of paper

Section 2 describes literature survey. Section 3 gives problem definition for system later section 4 describes System architecture and modules in system i.e. cloud service, video transformation engine and video optimization engine. Section 5 gives experimental results. Section 6 summarizes conclusion and future scope.

2. LITERATURE SURVEY

Smartphone has advanced OS which can be useful to access many applications. As compared to mobile phones, smart phones have poor battery life. The energy consumption of such smart phones is major challenge of this era. Hence study of work done in this area is quiet booming and interesting. Dynamic Tone Mapping: The proposed algorithm takes advantage of visual system features and tries to minimize the distortion occurrence between the brightness values of the individual pixels in the original frame and those of the backlight (optimized) scaled frame. Moreover, the proposed algorithm based on tone mapping is willing to respond to the highly efficient hardware and does not require any information regarding the histogram of the displayed frame, which makes it compatible with video applications [3]. Limitation: It saves energy but there is some distortion which can be caught by human visual system.

Backlight Dimming in Power- Aware Mobile Displays: The video distortion comprises of (i) spatial (intra-frame) distortion component due to image-sensitive backlight scaling (ii) temporal (inter-frame) distortion component due to large backlight dimming through multiple frames and modulated by characteristics of the human visual system. In this technique, the spatial distortion i.e. intra-frame distortion is upper-bounded by a user specified maximum allowed value. The second cause of distortion is the change in the luminance of the backlight-optimized video compared to the original video, i.e. the temporal distortion [5].It proposes a temporally-aware backlight scaling scheme for video, based on a detailed video distortion metric, which account for the intra-frame distortion and uses power density of the average luminance of different frames to measure the flickering effect by the backlight optimization [5]. Limitation: Sudden backlight level switching causes inter frame brightness distortion, which can cause bad perception on human visual system.

Dynamic backlight optimization using Proxy server: The proxy server middleware dynamically accepts the streaming video content such that brightness and contrast compensation and communicates control information i.e. operating backlight levels through the communication manager module to the client middleware. The video stream request is sent by client through the control stream to send to the proxy, with the current backlight level and device specific information. The proxy server uses the control stream to direct the client mobile device to set a new backlight level while playing streaming video. The proxy server maintains a database which has the videos available at the server, along with information specific to handheld device types [6]. There are three middleware adaptation policies that use the compensation algorithm.1) Simple backlight compensation 2) Constant backlight with video luminosity compensation 3) Dual-compensation approach [6]. Algorithm has used third approach which has maximum control on operating system, which requires both proxy and operating system interfaces for optimal operation [6]. Limitation: To overcome previous limitation in implemented techniques the algorithm has grouped images to change the backlight levels for that particular group. Grouping is done frame-vise not scene-vise, so grouping images can have different backlight levels for two images belonging to same scene which can be predicted by user.

Quality Based Optimization: It defines induced distortion value which is closest to a specified threshold as the optimized backlight dimming factor. It replaces the factor with the real backlight level Alfa, Alfa = N (N is the number of backlight levels) and the optimized backlight Dimmed level is represented as Alfa [2]. It illustrates quality distortion in terms of MSE (Mean Square Error) over different backlight levels. As Alfa increases, video quality distortion due to the brightness saturation decreases. Hence, for a given distortion threshold, it can find a unique Alfa for each image. In video applications, different frames may have distinct Alfa, it depends on the luminance of that particular frame. However, it is hard to design an accurate analytical representation of the quality distortion using Alfa as a parameter. Therefore it has adopted an optimized search based-approach, where it calculates the MSE (Mean Square Error) distortion with different Alfa values unless and until the specified distortion threshold is met [2]. Limitations: It does optimization without considering overall frames in the video, so there can be chance of having better optimization using improved technique.

3. PROBLEM DEFINITION

Following section represents scaling constraints which can avoid video distortion, flickering effect and adverse effect on user's perception, including h/w limitation along with power model. A video can have N frames F = f1, f2, f3..... fN which are displayed in sequence with constant rate. Every frame is matrix of pixel and each pixel has its own luminance known as pixel luminance which can has effect on perceptual luminance of pixel. Some papers have given relationship between perceptual luminance and backlight level i.e. perceptual luminance = pixel luminance * backlight level. Pixel luminance does not affect battery consumption greatly but backlight level surely does. Hence if we are lowering down backlight we need to increase pixel luminance so that perceptual luminance does not vary widely. Also every image frame has threshold value to lower down the backlight level. That threshold is the critical level (C (i)). Backlight can't be lowered down below its critical level, because beyond critical level contents on the frames are not visible. This can limit the lowest level of backlight for each frame. Element of visual perception: Light source of intensity I and change in I is I. If I is not bright enough, it can't be recognized by user; but if delta I is strong then subject can catch intensity change. The ratio of change of intensity to the actual intensity should remain constant; this is known as weber's law. To avoid adverse effect on user's perception weber's constant should be small or less than specified value. H/w of smart phone takes little amount of time to react with backlight change. We can't keep on incrementing or decrementing backlight level for each frame. Energy consumed by display system of smart phone is directly correlated to backlight level and backlight level is directly proportional to the energy consumed. Hence backlight level increases energy consumed i.e. battery consumption increases. Our objective is to scale down backlight level while playing video streaming application so that energy consumed to play video gets reduced excluding adverse effect on user's perception. While doing so we need to keep in mind some constraints before scale down the backlight 1) The backlight level applied to any image frame is not lower than the frames critical backlight level 2) The magnitude of a backlight change should not be greater than the product of current backlight level and differential ratio 3) The number of image frames between any two backlight changes is not less than a specified number [1].

4. SYSTEM ARCHITECTURE

This section presents a cloud-based energy saving service (EAAS), called the dynamic backlight scaling service, which minimizes the backlights energy consumption when displaying video streams on mobile devices.

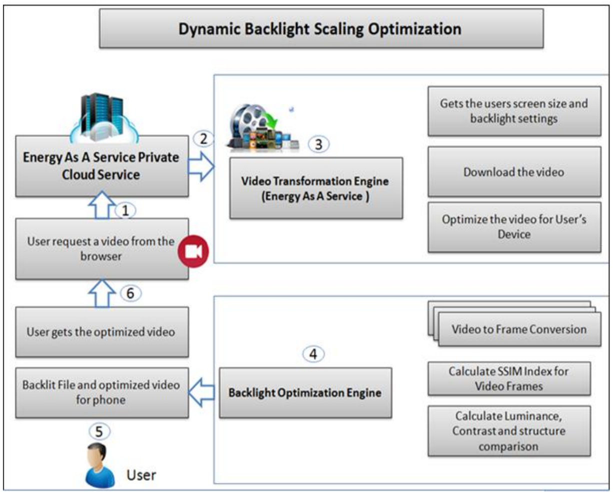

Figure 1

A possible way to realize the service would be to develop this as a value-added service offered by Internet service providers. Which will do conversion part on behalf of mobile phone. The service is presented in a way that is easy for end users to understand. They do not need to know how the service is provided and where the system that delivers the service is located. User just needs to use it through an application; service is running at back end.

4.1 Algorithm

1) Video framer: calculate frame rate for video

(frame rate i = no of frames/second).

2) RecordReader (FirstphaseofMapReduce method) conversion to (Key and value format).

3) FFMPEGFrameGrabber class has grapFrame() method which will access video frame by frame.

4) SSIM index between first frame (0) and last frame (i) of slot is calculated.

5) If SSIM index is greater than 0.6 Then apply same backlight level to each frame.

6) If else SSIM index between first frame (0) and (i/2) frame is greater than 0.6 Then apply same backlight level to frames from 0 to (i/2) and calculate backlight level for frames from (i/2+1) to (i).

7) Else calculate backlight level for frames from 0 to (i/2) and apply same backlight level to frames from (i/2+1) to (i)

8) New backlight file and video is sent to user

4.2 Calculations for SSIM

Index SSIM (Similarity Structure Index Matrix) index is measure of similarities between two image frames.

4.2.1 Luminance Calculations

$$\mu_x = \frac{1}{N}\sum_{i=0}^{N-1} xi$$

$$\mu_y = \frac{1}{N}\sum_{i=0}^{N-1} yi$$

$$l(x,y) = \frac{2\,\mu_x\mu_y + c1}{\mu_x^2 + \mu_y^2 + c1} \quad where\ c1 = 0.01$$

Where μ(x) and μ(y) represent the means of the original and optimized frames, respectively

4.2.2 Contrast Calculations

$$\sigma_x = \left(\frac{1}{N-1}\sum_{i=0}^{N-1}(x_i - \mu_x)^2\right)^{1/2}$$

$$\sigma_y = \left(\frac{1}{N-1}\sum_{i=0}^{N-1}(y_i - \mu_y)^2\right)^{1/2}$$

$$c(x,y) = \frac{\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad where\ C2 = 0..02$$

Where σ(x) and σ(y) represent the standard deviations of each of the signals

4.2.3 Structure Calculations

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=0}^{N-1}(x_i - \mu_x)(y_i - \mu_y)$$

$$s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x \sigma_y + c_3} \quad where\ C3 = 0.03$$

σ_{xy} is the covariance of the two frames.

4.2.4 SSIM Index

$$SSIM\,(x,y) = [l(x,y)] * [c(x,y)] * [s(x,y)]$$

where $\alpha = \beta = \gamma = 1$

$$SSIM(x,y) = \frac{(2\,\mu_x\mu_y + c1)(2\sigma_{xy} + c_2)}{(\sigma_{xy} + c_3)(\sigma_{xy} + c_3)}$$

5. EXPERIMENTAL RESULTS

We have done the comparative study of three algorithms viz. Mean square error, Color histogram and SSIM , after performing analysis, we found that SSIM Index gives the maximum backlight adjustment probabilities, so that P() (power model) can be reduced further. Figure 2 shows the comparative study of three algorithms.

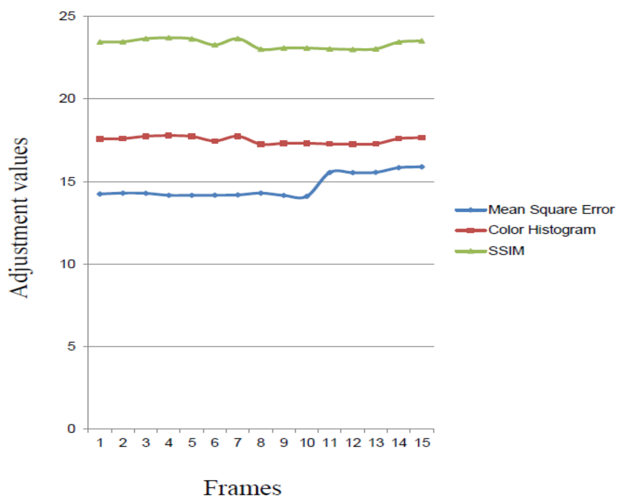

Figure 2

Experimental results shows that energy consumption while playing video depends on the brightness of video, if video is bright energy saving is maximum, but if video is dark we can't reduce its backlight greatly, so energy saving is minimum. We have three videos for experiment viz. Mortak Kombat X Official trailer: Animated movie, IBN Lokmat.mp4: news video, CABIN-FEVER.mp4: Hollywood movie which have different parameters as shown in figure 3. Figure 4 shows original brightness level of each frame.

Video Name	Frame Rate	Total Frames	Time in Sec
Mortal Kombat X official trailer.mp4	30	3893	129
IBN_Lokmat.mp4	25	3397	135
CABIN_FEVER Movie.mp4	23	2557	111

Figure 3

Frame	Mortal Kombat X official trailer.mp4	IBN Lokmat.mp4	CABIN_FEVE R Movie.mp4
1	10.37275505	203.3511963	30.39486504
2	11.80592632	212.0909271	35.86068726
3	14.53609276	196.6169586	54.05879974
4	16.30043411	192.3379822	78.48503113
5	17.91634941	191.5857239	104.1830215
6	17.2821579	132.6075592	77.91508484
7	18.64741135	123.9704437	76.56473541
8	20.57975197	134.4868469	50.75155258
9	23.57698441	149.1856232	44.35792923
10	25.6813736	151.8177032	52.28021622
11	26.22992401	154.509433	56.41465213
12	27.78628771	153.5465851	68.67455391
13	29.3426514	148.5837372	69.9344557
14	30.8990151	148.6208893	71.19435748
15	32.45537879	151.6580414	72.45425927
16	34.01174249	145.6951935	73.71416106
17	22.56810618	144.7323456	74.97406284
18	22.12446988	146.7694977	76.23396463
19	23.68083357	140.8066498	77.49386642
20	22.23719727	138.8438019	78.7537682
21	23.79356096	139.880954	80.01366999
22	23.34992466	132.9181061	81.27357178
23	24.90628836	131.9552582	82.53347356
24	25.46265205	129.9924103	83.79337535
25	27.01901575	125.9704376	85.05327714
26	30.57537944	124.9332855	86.31317892
27	31.13174314	123.8961334	87.57308071
28	32.68810683	132.8589813	88.8329825
29	33.24447053	141.8218292	90.09288428
30	32.80083422	135.7846771	91.35278607

Figure 4

Figure 5 shows the backlight level after adjustments. These brightness values when compared to the brightest level we can calculate the optimization levels and energy saving by profit-loss formula. These results are given in figure 6.

6. CONCLUSION AND FUTURE WORK

With the help of an algorithm, power consumed by android devices while playing multimedia streaming application can be reduced to 10-50 percent (depending upon the video). System is trying to maintain user perception and avoiding video distortion while scaling down the backlight. So user can't predict the difference between optimized video and original video. Green computing can be achieved by the system as energy is being saved. In future, this algorithm can be implemented at patch level, which means at operating system level so that smart phones will by default have backlight optimizer in it. Also application can be made hybrid so that it can run on different operating systems like blackberry, I phone, windows etc.

Serial No	Mortal Kombat X official trailer.mp4	IBN_Lokmat.mp4	CABIN_FEVER Movie.mp4	MAX_W/O-Optimize
1	245	52	225	255
2	243	43	219	255
3	240	58	201	255
4	239	63	177	255
5	237	63	151	255
6	238	122	177	255
7	236	131	178	255
8	234	121	204	255
9	231	106	211	255
10	229	103	203	255
11	229	100	199	255
12	227	101	186	255
13	226	106	185	255
14	224	106	184	255
15	223	103	183	255
16	221	109	181	255
17	232	110	180	255
18	233	108	179	255
19	231	114	178	255
20	233	116	176	255
21	231	115	175	255
22	232	122	174	255
23	230	123	172	255
24	230	125	171	255
25	228	129	170	255
26	224	130	169	255
27	224	131	167	255
28	222	122	166	255
29	222	113	165	255
30	222	119	164	255
Total Energy Used	6917	3168	5468	7650
Total energy Saved	9.581788487	58.58600276	28.51668416	0

Figure 5

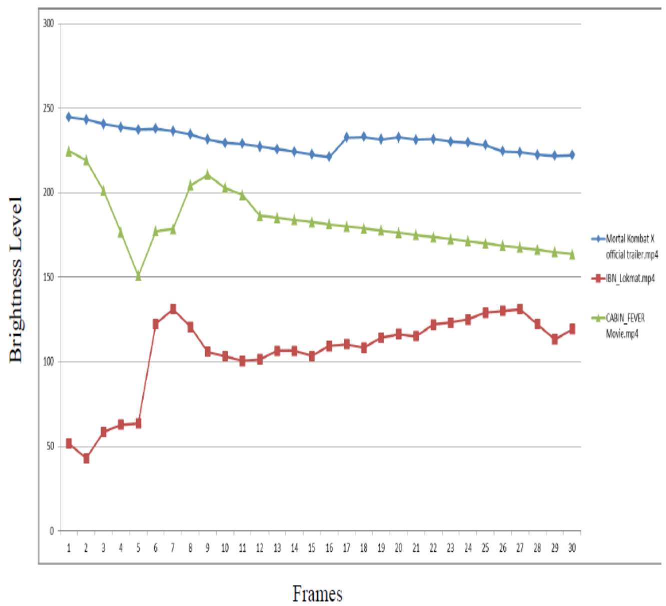

Figure 6

REFERENCES

[1] Chun-Han Lin, Member, IEEE, Pi-Cheng Hsiu, Member, IEEE, and Cheng-Kang Hsieh, Student Member, IEEE, Dynamic Backlight Scaling Optimization: A Cloud-Based EnergySaving Service for Mobile Streaming Applications. IEEE Transaction on computers, VOL. 63, NO. 2, Feb 2014

[2] N. Chang, I. Choi, and H. Shim, DLS: Dynamic Backlight Luminance Scaling of Liquid Crystal Display. IEEE Trans. Very Large Scale Systems, vol. 12, no. 8, pp. 837-846, Aug. 2004.

[3] 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, vol. 2007, p. 4, 2007.

[4] W.-C. Cheng and C.-F. Chao, Minimization for LED-backlit TFTLCDs. Proc. ACM/IEEE 43rd Design Automation Conf. (DAC), pp. 608-611, 2006.

[5] C.-W. Huang and S.-L. Tsao, Minimizing Energy Consumption of Embedded Systems via Optimal Code Layout. IEEE Trans. Computers, vol. 61, no. 8, pp. 1127-1139, Aug. 2012.

[6] A. Iranli, H. Fatemi, and M. Pedram, HEBS: Histogram Equalization for Backlight Scaling. Proc. ACM/IEEE Conf. Design, Automation and Test in Europe (DATE), pp. 346351, 2005.

[7] A. Iranli, W. Lee, and M. Pedram, Backlight Dimming in Power- Aware Mobile Displays. Proc. ACM/IEEE 43rd Ann. Design Automation Conf. (DAC), pp. 604-607, 2006.

[8] A. lranli and M. Pedram, DTM: Dynamic Tone Mapping for Backlight Scaling. Proc. ACM/IEEE 42nd Ann. Design Automation Conf. (DAC), pp. 612-616, 2005.

[9] D. Niu, Y. Chen, and Y. Xie, Low-Power Dual-element Memristor Based Memory Design. Proc. ACM/IEEE Intl Symp. Low-Power Electronics and Design (ISLPED), pp. 25-30, 2010.

[10] S. Pasricha, M. Luthra, S. Mohapatra, N. Dutt, and N. Venkatasubramanian, Dynamic Backlight Adaptation for LowPower Handheld Devices. IEEE Design Test of Computers, vol. 21, no. 5, pp. 398-405, Sept./Oct. 2004.

[11] Facebook,http://www.facebook.com,2015

[12] YouTube. http://www.youtube.com, 2015.

[13] Google http://Google.co.in, 2015

BIOGRAPHIES

Bhavika Lahoti currently pursuing ME Computer Engineering from MITCOE,Pune Graduated BE Computer Science from MIT pune in 2013.

Prof. Hanumant Pawar has completed his full-time M.E IT MIT, Pune and B.E. IT from MIT Pune, both under the S. P. University of Pune. He has eleven plus years' experience of teaching. His area of interest is Artificial Intelligence.