

# Too Many Pixels to Perceive: Subpixel Shutoff for Display Energy Reduction on OLED Smartphones

Zhisheng Yan  
Georgia State University

Chang Wen Chen  
State University of New York at Buffalo

## ABSTRACT

Organic light-emitting diode (OLED) has been widely recognized as the next-generation mobile display. Recently, smartphone manufacturers have been pushing up the pixel density of OLED display. Unfortunately, such an effort does not necessarily improve the everyday viewing because of the limitation in human visual acuity. Instead, high pixel density OLED can drain the battery power even more quickly since the power dissipation of OLED is determined by the number of displayed pixels and their RGB values, or subpixels. This paper presents a new design dimension to remedy this prevailing issue by leveraging the intuition that shutting off redundant subpixels of the display content on OLED can reduce power consumption without impacting viewing perception. We introduce ShutPix, a power-saving display system for OLED smartphones that can optimally shut off the redundant subpixels before the content is displayed. Inspired by the motivational studies, ShutPix is empowered by a suite of designs based on visual acuity, human perception, and content redundancy. Experimental results show that ShutPix can, on average, reduce 21% of display power and 15% of system power without degrading user viewing experience.

## CCS CONCEPTS

• Information systems → Mobile information systems;

## KEYWORDS

OLED display; energy reduction; smartphones

## ACM Reference format:

Zhisheng Yan and Chang Wen Chen. 2017. Too Many Pixels to Perceive: Subpixel Shutoff for Display Energy Reduction on OLED Smartphones. In *Proceedings of MM'17, Mountain View, CA, Oct., 9 pages*. DOI: 10.1145/3123266.3123344

## 1 INTRODUCTION

Organic light-emitting diode (OLED) is recently emerging on most newly released smartphones. While the energy efficiency of OLED hardware is advanced slowly, there appears an increasing resolution and pixel density on OLED smartphones. This effort has been arguably marketed for providing sharper multimedia content on mobile display. Unfortunately, except for virtual reality (VR) applications with unusual viewing distance [24], such an upsurge in pixel density does not necessarily improve users' everyday viewing

experience in regular use cases. Instead, it will even drain much more display power and degrade user experience [27].

In particular, most manufacturers have pushed up their OLED smartphone resolution to 2560×1440 (2K), which boosts the display pixel density to more than 500 *Pixels Per Inch (PPI)*. Despite these impressive statistics, a well-known issue is that users are unlikely to differentiate such subtle content detail in normal usage because of the limitation in human visual acuity. According to an image viewing study [26], one half of the subjects identified a 577-PPI OLED phone as a better display while the other half preferred a same-size phone with 471 PPI. The reason for this controversy is that the extremely high PPI makes each pixel shrunk to an indistinguishable size and thereby similar image detail are observed on both displays. In fact, ~300 PPI has been recognized as the golden number for users with normal vision in everyday viewing [3] and are still adopted by a few models (e.g., iPhone 7) (Section 2).

More importantly, the growing OLED pixel density will even mitigate the display energy efficiency. This is because the display power of self-emissive OLED is determined by the total number of displayed pixels as well as the values of their corresponding RGB components [11], i.e., *subpixels*. Under the same display size, higher pixel density will definitely consume more battery power. Based on our own measurement, an OLED smartphone of 518 PPI causes up to 48% more per-inch display power than a device of 403 PPI and its display power can take up 58% of the system power (Section 3).

The excessive amounts of pixels and the resulting OLED energy has not been studied systematically. In this paper, we bridge this gap by leveraging the intuition that shutting off redundant tiny subpixels of the display content on OLED can save battery power, but will not impact the user perception. We aim at maximally shutting off subpixels on high-PPI OLED smartphones. It is important to note that unlike traditional color transform schemes for power-saving OLED that treat every RGB subpixels *necessary* and fine-tune their *signal magnitude* [6, 8, 10, 15, 17, 34], subpixel shutoff creates a new research direction in identifying the *redundant* subpixels and thus reducing the *spatial frequency* of the display signal. It is the decreasing number of redundant subpixels that brings the power reduction. From this perspective, the redundant subpixel shutoff is complementary to existing color transform schemes.

However, turning this promising strategy into practice requires us to overcome two daunting challenges. First, shutting off subpixels may damage the display appearance, introducing incomplete content such as discontinuous lines or objects. The color perception of the content may also change since the human eye will smooth out the color of neighboring on/off pixels. Second, due to the varying content features, different regions of the display shall exhibit different visual redundancy and therefore require a region-specific design. This is further complicated by that OLED power is also content-dependent, making it hard to strike a perfect balance.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

MM'17, Mountain View, CA

© 2017 ACM. 978-1-4503-4906-2/17/10...\$15.00

DOI: 10.1145/3123266.3123344

To tackle these challenges, we propose *ShutPix*, a display power-saving system for OLED smartphones that can optimally shut off (set value to zero) redundant subpixels before the content is displayed. Based on visual acuity and a set of motivational studies, ShutPix derives the minimum perceivable region and candidate patterns for subpixel shutoff. Furthermore, we explore the luminance masking and contrast masking effects, and propose a visual redundancy based weighting scheme to differentiate display regions. Finally, ShutPix enforces the subpixel shutoff by optimizing the shutoff pattern for each region in order to minimize the overall display power without affecting viewing perception.

We have implemented ShutPix on Android devices. ShutPix can support legacy Apps for everyday viewing and can be disabled for VR. We validate ShutPix by real-world experiments under various practical settings including App type, display content, and OLED device, as well as by comparing ShutPix with prior color transform schemes. The results show that ShutPix can, on average, save 21% of display power and 15% of system power without degrading user viewing experience.

In sum, the contributions of the proposed research include:

- A set of motivational studies that inspire the design requirements of ShutPix. (Section 3).
- A systematic design of ShutPix governed by the unique features of OLED devices and human vision. (Section 4).
- A practical demo of subpixel shutoff to reduce power without impacting users' visual perception. (Section 5-6)

## 2 BACKGROUND AND RELATED WORK

### 2.1 Human Visual Acuity

Visual acuity is a measure of the spatial resolution of the human vision system that is dependent on both the object size and the viewing distance. A normal vision is defined as the ability to discriminate two objects by 2 arc minute, i.e., 1/30 degree [21]. As shown in Figure 1, the discriminating angle  $\theta$  for objects can be defined as,

$$\theta = 2 \arctan\left(\frac{l}{d}\right) \quad (1)$$

where  $l$  is the size of one pixel and  $d$  is the object-to-eye distance.

According to (1), at a typical smartphone viewing distance of 12 inches [4], the minimum size of the two objects that a normal person can distinguish is  $89 \mu\text{m}$ . As for OLED display, a normal person can perceive the content details that vary in two pixels per  $\theta$  at the pixel size of  $89 \mu\text{m}$ . This is equivalent to only 286 pixels per inch (PPI). That is why a popular rule-of-thumb number for mobile pixel density is 300 PPI [3], as marketed in Apple's "retina" display.

### 2.2 Smartphone Display and OLED Power

The traditional LCD display is illuminated by an external backlight in the display panel. Hence, by scaling the backlight, one can reduce the display power at the expense of the uniformly decreased screen brightness [19, 40]. In contrast, the emerging OLED displays are self-emissive. Each pixel on the OLED is defined by the values of red, green and blue subpixels. Since RGB subpixels have different illuminance efficacies when displaying different values, the power consumption of OLED is directly decided by the display content. Therefore, the display power of an OLED smartphone has been

widely expressed as [1, 7, 11, 18]

$$P_{disp} = \sum_{i=1}^N f_r(r_i) + \sum_{i=1}^N f_g(g_i) + \sum_{i=1}^N f_b(b_i) \quad (2)$$

where  $N$  is the number of pixels and  $r_i, g_i, b_i$  are the RGB values of pixel  $i$ . The power models for displaying RGB subpixels ( $f_r, f_g, f_b$ ) has a power-law increasing trend with the RGB values. Furthermore, when  $r_i = g_i = b_i$ , the display power of blue subpixel is the highest whereas red and green subpixels have similar power efficiency.

### 2.3 Related Work

**OLED power reduction.** On commercial devices, brightness dimming [32, 39] that uniformly darkens the OLED screen has been implemented as a default setting. Furthermore, researchers have strived to improve OLED energy efficiency by color transform. Chameleon [10] transformed the color map of web browsers to deliver power-efficient color based on user-defined preference. Focus [35] exploited users' top-down reading manner on smartphones and proportionally dimmed the screen to save display power. Although these systems can result in large power reduction, they concentrate on preserving the usability rather than the fidelity of display and thus may not be suitable to Apps with rich media content.

To improve content fidelity, Anand *et al.* mapped each pixel color under the constraints of hue, saturation and vale (HSV) [1]. Chen *et al.* classified video content such that the HSV thresholds of color transform were decided by the video category [8]. Lin *et al.* dimmed each displayed image regions individually subject to the structural similarity (SSIM) constraint [18]. Kang *et al.* converted the luminance histogram to simultaneously increase luminance contrast and reduce OLED power [15]. Crayon [34] transformed the color and shape of objects on the display under color distance threshold. These works focus on manipulating the color by only considering the content fidelity. Instead, ShutPix additionally explores the device characteristics, i.e., the substantial gap between pixel density and visual acuity, to obtain another power-saving space.

Both color transform and ShutPix are special types of tone mapping. Based on the fidelity constraint, color transform fine-tunes the signal magnitude of every subpixel without setting it to zero. In contrast, based on the excessive PPI, ShutPix steers to a novel direction of reducing the signal spatial frequency by only shutting off redundant subpixels. Without degrading perception, ShutPix can achieve complementary power saving over color transform.

**Hardware-level subpixel rendering.** In order to render pixels with high quality and low power, manufacturers have carefully configured the internal hardware layout for the display panel. While some displays employ more than three hardware units to render a single pixel, e.g., red, green, blue and white (RGBW) [33], some others overlap the hardware units, e.g., red and blue diodes are

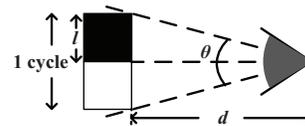


Figure 1: Sketch of visual acuity for normal vision.

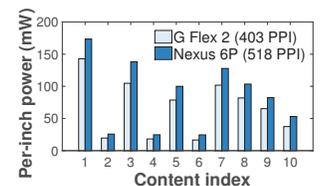


Figure 2: Higher PPI leads to higher power consumption.

shared when rendering two neighbor pixels in PenTile RGBG [12].

In this paper, however, our design is a software approach. We deal with RGB subpixels of the display content, which are defined by the bitmap. This is agnostic to how RGB subpixels are actually rendered in the hardware. To shut off a subpixel, we set the corresponding R/G/B value to zero and send the modified bitmap data to the hardware. No matter which hardware layout is used, the perceived color of a rendered pixel will be only dependent on the RGB values.

### 3 MOTIVATIONAL STUDIES

In this section, we conduct a set of studies that directly motivate the design choices in ShutPix.

#### 3.1 Impacts of Pixel Density

**OLED display power.** To study the impacts of pixel density on OLED power, we display the same content in full screen on a Huawei Nexus 6P with 518 PPI and a LG G Flex 2 with 403 PPI. We test 10 pieces of 2K content (natural image/App screenshot). To measure the display power, we follow the methodology in [10, 11]. We first measure the system power when the display shows a pure black image, where the reading indicates the background power. We then subtract the background power from the system power when displaying a given image and thus derive the OLED display power. To measure the system power, we use Qualcomm Treppn, a tool that directly reads the hardware data. The test devices have been listed by Qualcomm as supported devices that provide accurate result [29]. Each measurement lasts for 120 seconds from the moment the content is displayed on the screen. We repeat the measurement for three times and the average power reading is reported.

The results of per-inch display power in Figure 2 show that Nexus 6P with higher PPI causes 22%~48% more display power across the images. This phenomenon can be easily explained by (2), i.e., a device with more pixels expends more power. Note that the power efficacy for displaying the same pixel is actually different for the two devices. For the same RGB value, Nexus 6P even dissipates a lower per-pixel power. However, due to the much higher pixel density, Nexus 6P still yields a substantially higher per-inch power. Furthermore, we find that OLED power can reach up to 58% of the system power. This is consistent with prior measurement [5] that has verified the significance of display power. Building on these results, we conclude that excessive PPI indeed brings about large OLED power consumption and it is *desirable* to reduce such power.

**User perception.** To investigate the user perception under different pixel density, we carry out a user study by following ITU stimulus-comparison method [22]. As suggested in [22], we recruit 25 users (16 males and 9 females, age 19 to 35) with normal or corrected vision via online forum or email list. We show them a pair of two devices displaying the same content and ask them if the content on Nexus 6P is different from that on G Flex 2. We instruct them to view the content in a casual and everyday manner without putting their eyes unusually close to the screen.

We observe that an average of 95% of users (90%~100% across images) do not see any difference on the two distinct PPIs. This confirms the limitation in human visual acuity. Therefore, we can conclude that it is *feasible* to improve energy efficiency by selectively shutting off pixels/subpixels as there is significant visual redundancy on modern OLED with over 300 PPI. Moreover, users

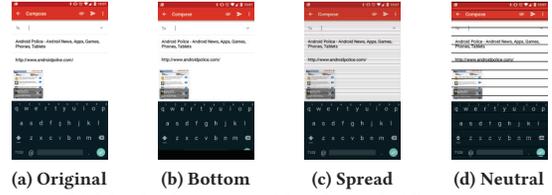


Figure 3: (b) blocks bottom of keyboard; (d) is too intrusive; (c) is voted as the most complete content among these three.

have expressed different perception towards different content. For example, users report noticeable difference only on smooth and plain surfaces, e.g., a human face, whereas they are not able to discern any difference on the extremely dark content with complex texture, e.g., a dark-green bush. This observation motivates a *content-specific* pixel/subpixel shutoff over the display regions.

#### 3.2 User Perception of Pixel/Subpixel Shutoff

As pixel/subpixel shutoff must be conducted without degrading user experience, a serial of user studies have been performed to examine how this technique impacts human perception.

**Spatial locations of pixel shutoff.** We first explore what are the best locations for pixel shutoff. We pick a typical 2K screenshot with various content features and colors across different image regions. Three spatial distributions are employed to shut off 1/15 of the pixels: (a) *Bottom*: Only the bottom part of the content is shut off in order to minimize the changed display area. (b) *Spread*: We select one shutoff location every 14 pixels in a vertical scan to minimize the change within a random area. (c) *Neutral*: a in-between method of (a) and (b) that chooses 4 shutoff locations every  $4 \times 14$  pixels to balance the overall completeness and random area completeness.

The screenshots in Figure 3 show the original content and processed content on a Motorola Nexus 6. We ask the 25 users to identify the most complete/continuous content compared with the original content. Consequently, all but two users vote Spread while two users choose Bottom. We conjecture that Spread spatially distributes the pixel shutoff over the entire image, balancing the overall usability and fidelity. In contrast, Bottom blocks part of the keyboard while Neutral is clearly too intrusive. The votes for Bottom may be because the users did not realize there are still keyboard buttons under the black area. From this study, we arrive at two remarks.

First, it is critical for ShutPix to *spatially distribute the shutoff locations as much as possible*. However, Spread strategy with uniform shutoff location is not sufficient because it still incurs discontinuous content, i.e., black horizontal lines across the display. The neighbor shutoff locations are seen connected such that the discontinuity is magnified. Hence, ShutPix should push the nearby shut off locations as far from each other as possible. Second, the color perception will be too dark if Spread strategy is adopted. In fact, we aggressively shut off all RGB subpixels for each shutoff location and these black pixels are completely different from the original colors. This observation motivates us to adopt a conservative strategy by shutting off *a subset of the RGB subpixels* at each location. Thus, a color more similar to the original color will be perceived.

**Color change of subpixel shutoff.** We now study what is the best subset of RGB subpixels to shut off within one pixel location given that the target locations are fixed by Spread strategy. Unlike

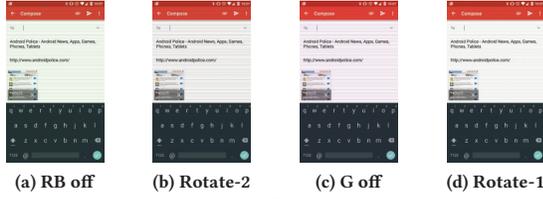


Figure 4: (a)(c) have totally different color; (b) is too dark; (d) is the most similar color appearance to original content.

Spread shutting the entire pixel, we shut off two subpixels identically at all locations, yielding three strategies: RG off, RB off and GB off. We also rotationally shut off RG, RB, and GB subpixels at each location (denoted by *Rotate-2*). Furthermore, four more conservative schemes that shuts off one single subpixel at all locations are compared, i.e., R off, G off, B off, and rotating the R/G/B off (*Rotate-1*). We observe in the user study that all users select the *Rotate-1* as the most similar content to the original, as exemplified in Figure 4. We now reach the following findings.

First, *only one subpixel should be shut off at each shutoff location*. In fact, after shutting off two subpixels at each location, the content becomes too dark to be accepted (Figure 4a–4b). This is because human eyes are much more sensitive to luminance than chrominance and shutting off two or more subpixels will largely decrease the display luminance. Second, *the number of each R, G, and B subpixels to be shut off should be as close as possible*. We can see that shutting off identical subpixels at all locations will change the overall color perception of the content considerably (Figure 4a and 4c). This is similar to the effects of a color filter, which will unavoidably undermine the overall color balance. Finally, although the results are acceptable, *Rotate-1* still incurs noticeable difference, both *discontinuous content* and *distorted color appearance*. By exploiting the user preference obtained from these studies, we will deliberately design ShutPix to overcome these two issues.

## 4 DESIGN OF SHUTPIX

In this section, we introduce the design of ShutPix based on the insights gained from the motivational studies.

### 4.1 Architecture

In a mobile operating system, an App can update the display content by either loading images or rendering graphic data [28]. The content will be resized, drawn, and enqueued into buffer queues before delivered to the graphic subsystem for GPU processing and frame composing. As shown in Figure 5, ShutPix intercepts the drawing calls and overrides the content with redundant subpixel shutoff.

Recall that incomplete content and color change have been observed as the two major issues of subpixel shutoff in the motivational studies. The first design principle of ShutPix is that although the human eye is not able to notice the content variation between two individual pixels on a high-PPI OLED, there is a minimum region that users may perceive incomplete content (Figure 5 spatial frequency setup). As long as the subpixel shutoff within every minimum perceivable region yields continuous content (Figure 5 pattern derivation), the completeness of the whole display will be

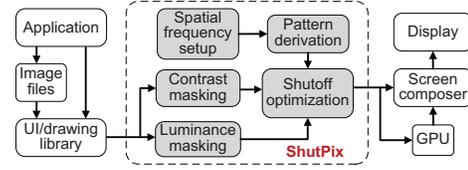


Figure 5: Architecture of ShutPix.

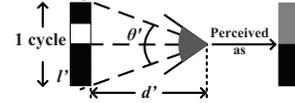


Figure 6: Pixel variation on high-PPI display cannot be seen.

preserved. Furthermore, ShutPix seeks the optimal subpixel shutoff pattern for each minimum perceivable region while satisfying the constraint of just noticeable difference (JND) in display color (Figure 5 shutoff optimization), as well as considering the different visual redundancy of color appearance in different content regions (Figure 5 contrast masking and luminance masking). In the following, we will introduce each ShutPix module in Figure 5.

### 4.2 Shutoff Spatial Frequency Setup

Based on human visual acuity and display pixel density, we now derive the minimum perceivable region of subpixel shutoff. The pixel density implies how much detail can be shown within a region. If it is too high, a person with normal vision will not differentiate the actual color variation and content detail at the per-pixel level. Instead, the user will only generalize the perception of multiple neighbor pixels. This is because human eyes can be approximated as a low-pass filter that removes high-frequency signal details [23].

Recall that at a typical viewing distance  $d = 12$  inches, a person with normal vision can perceive the content variation between 2 pixels at a pixel size  $l = 89\mu\text{m}$  [21]. For an OLED smartphone with higher pixel density, the pixel size  $l'$  becomes much smaller than  $l$  such that there are more than 2 pixels within the discriminating angle. Thus, the pixel variation within the discriminating angle tends to be more detailed than a normal vision can support and such content variation will not be perceived. As shown in Figure 6, we cannot see the black and white transition at the physical pixel. However, we can perceive the gray-black transition that is smoothed from the physical pixels. The minimum cycle that one may perceive a content variation now becomes 4 physical pixels per  $\theta'$  for this example. Note that the perceived color might also change. We will handle this in the following sections. Formally, for a high-PPI display, we can derive the minimum perceivable number of pixels,  $n$ , by enforcing that the discriminating angle  $\theta'$  for the given display should be approximated as standard  $\theta$  ( $\frac{1}{30}$  degree), i.e.,

$$\begin{aligned} \theta' &= 2 \arctan \frac{\frac{n}{2}l'}{d'} \approx \theta \\ \Rightarrow \frac{\frac{n}{2}l'}{d'} &\approx \frac{l}{d}, \Rightarrow n \approx \frac{2ld'}{l'd} \end{aligned} \quad (3)$$

where  $l'$  ( $l' < l$ ) is the tiny pixel size on high-PPI display (derived from resolution and PPI) and  $d'$  is the new viewing distance.

By extending this derivation to two-dimensional display content, we can acquire the minimum perceivable region in  $n \times n$  pixels, which is herein called *block*. As long as we guarantee that the subpixel shutoff for every block does not induce any discontinuity

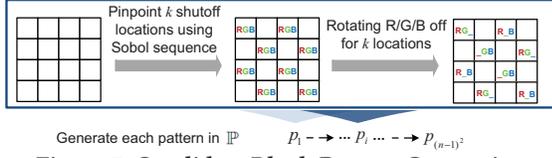


Figure 7: Candidate Block Pattern Generation.

content, we can then shut off subpixels in a block by block fashion, i.e., a spatial frequency of every  $n \times n$  pixels, and consequently maintain the content completeness of the whole display.

### 4.3 Block Shutoff Pattern Derivation

We then design the block shutoff pattern, i.e., which subpixel at which location of the  $n \times n$  block can be shut off.

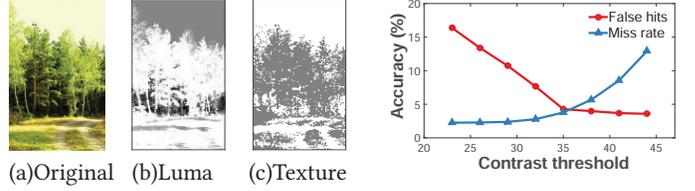
Ideally, we should test all possible patterns of subpixel shutoff within a block to acquire the optimal choice. However, even for a display with the smallest  $n = 3$ , there are  $(2^3)^{3 \times 3} \approx 10^9$  choices for one single block. Repeating this computation for all blocks is impractical for real-time display. Thereby, we propose to pre-design a much smaller set of candidate patterns for subpixel shutoff within a block to allow ShutPix to search the optimal pattern online.

We pre-design candidate patterns with different levels of power saving, indicated by the number of subpixels to shut off. At the same time, we seek to minimize the perception change for each pattern. The generating algorithm of block shutoff patterns is depicted in Figure 7. Given a  $n \times n$  block, we generate a set of patterns  $\mathbb{P} = (p_1, p_2, \dots)$ . For these patterns, the number of subpixels to shut off varies from  $n$  to  $n(n-1)$  (totally  $(n-1)^2$  patterns) in order to avoid either insufficient power saving or excessive perception change. Building on the motivational studies, it is critical to distribute the shutoff locations within a block in order to prevent clustered subpixel shutoff and to balance the content completeness. We thus propose a Sobol sequence based scheme to find evenly distributed locations. Sobol sequence is a low-discrepancy sequence that can approximate the even partition on a unit interval [14]. The sequence is generated such that successive points at any stage are as far away as possible from previously generated points and can be extended to two dimensions to represent the pixel coordinates. Note that a simple random sequence will not work since every time to decide a random location, all points would enjoy the same probability even if they are close to existing shutoff points.

According to the insights in Section 3.2, we shut off only one single subpixel (R or G or B) for each pinpointed location to keep the color change at a minimum. Furthermore, RGB subpixels are shut off in a rotating fashion at these locations to maximally avoid continuous shutoff of one particular subpixel. As a result, the total number of R, G, and B subpixels to be shut off should be approximately the same. If this cannot be guaranteed, blue subpixel is prioritized to shut off because it is the most power-hungry subpixel and it has the least impacts on content luminance to which human eyes are more sensitive than content chrominance [20].

### 4.4 Luminance Masking Based Weighting

As discussed in Section 3.1, viewing perception is content-dependent. Human eyes will manifest different levels of just noticeable difference (JND) [9, 41] in different display regions. Therefore, we divide the display into  $m \times m$  macroblocks and strive to obtain the weighting effects of visual redundancy for each macroblock. We introduce



(a)Original (b)Luma (c)Texture

Figure 8: Example weight map: (b) highly bright/dark regions -> lower weights (in gray); (c) complex texture -> lower weights (in gray) Figure 9: Overall accuracy of the proposed complex texture detection.

macroblock, which contains multiple  $n \times n$  blocks, because the content features normally stay unchanged within a block with only several tiny pixels. This can boost the computation efficiency of ShutPix by reducing the weighting granularity.

One important visual redundancy phenomenon is luminance masking. It describes that the extremely dark or bright background tends to “blind” the eyes. This reduces the vision sensitivity and increases the JND threshold of color change. By adopting the sensitivity function in [36] and normalizing it into [0,1], we propose the luminance masking weight,  $W_{lum}$ , for each macroblock as,

$$W_{lum} = \begin{cases} 1 - \frac{60 - Y_{avg}}{150} & Y_{avg} \leq 60 \\ 1 & 60 < Y_{avg} \leq 170 \\ 1 - \frac{Y_{avg} - 170}{425} & Y_{avg} > 170 \end{cases} \quad (4)$$

where  $Y_{avg}$  is the average luminance of the macroblock. When a macroblock is extremely dark or bright, it is weighted less, indicating that human eyes are less sensitive to the content and more subpixels may be shut off. We plot an example weight map for luminance masking using the proposed weighting method. The relatively dark and bright content of the original Figure 8a have smaller weights (gray dots) in the weight map in Figure 8b, whereas the content with regular luminance becomes white in the weight map. This implies that the proposed weighting method is indeed effective in modeling luminance masking.

### 4.5 Contrast Masking Based Weighting

Another JND effect we explore is contrast masking, which refers as the fact that noise is difficult to be noticed in complex texture regions while easily noticeable in regions showing smooth content. This provides us the opportunity to weigh less the complex texture regions in accordance to their higher visual redundancy. Existing works detect complex texture regions by subband transform, e.g., wavelet domain. However, this can introduce heavy computation and delay in smartphone display. Therefore, we propose a simple yet accurate algorithm to classify the macroblocks into complex texture or smooth texture. We compute the maximum pixel difference in the four sub-squares of a macroblock and calculate the average of them as the contrast  $C$  of this macroblock, i.e.,

$$C = \frac{1}{4} \sum_S (\max_{x,y \in S} Y(x,y) - \min_{x,y \in S} Y(x,y)) \quad (5)$$

where  $S$  is the top left, top right, bottom left, and bottom right sub-squares of a macroblock,  $Y(x,y)$  is the luminance of pixel  $(x,y)$ . Note that it is sufficient to only include luminance component in (5) since texture is not impacted by chrominance. If the contrast

is smaller than a threshold  $C_{th}$ , the macroblock is detected as a smooth macroblock and its contrast weight  $W_{con}$  is kept at one, i.e., not prioritized for subpixel shutoff. Otherwise, it is treated as complex texture and the contrast weight is penalized by half, as effectively used in JND-based video coding [38].

The key challenge is to determine the contrast threshold  $C_{th}$  in the proposed detection scheme. We pull a dataset of 30 screenshots/images at  $240 \times 352$ . We configure the macroblock as  $16 \times 16$  pixels, which is the default setting in image/video standards (JPEG [30] and H.264 [37]). We then manually label the texture type of each macroblock, producing a total of 9900 ground truth samples. We run the detection algorithm on the dataset using various  $C_{th}$  and show the overall accuracy in Figure 9, wherein false hit is the ratio of falsely detected complex texture to the number of macroblocks and miss rate is the percentage of missed complex texture. A greater  $C_{th}$  produces less complex texture for subpixel shutoff and thus better maintains the display quality. However, it may also miss some visual redundancy and drop the power reduction. Given the stable trends of both curves, we can safely determine the default  $C_{th}$  as 35 (with false hit and miss rate below 5%). By employing the optimal  $C_{th}$ , we can observe in the weight map in Figure 8c that the proposed scheme can accurately detect complex texture (gray dots) and thereby ensure more exploration of visual redundancy.

#### 4.6 Subpixel Shutoff Optimization

To optimize ShutPix, it is necessary to select an optimal shutoff pattern for each block (minimum perceivable region) such that the display power is minimized and the human perception of luminance and chrominance perception is maintained. In other words, we must apply the JND constraint of subpixel shutoff on the content.

We present the optimization that searches among the shutoff pattern set  $\mathbb{P}$  for each block. The problem shall minimize the display power  $P_{disp}$  subject to the weighted luminance and chrominance change, i.e.,

$$\begin{aligned} \min_{p \in \mathbb{P}} \quad & P_{disp} \\ \text{s. t.} \quad & W_{lum} W_{con} \begin{bmatrix} |Y - Y(p)| \\ |Cb - Cb(p)| \\ |Cr - Cr(p)| \end{bmatrix} \leq \begin{bmatrix} \gamma_Y \\ \gamma_{Cb} \\ \gamma_{Cr} \end{bmatrix} \end{aligned} \quad (6)$$

where  $Y, Cb, Cr$  are the average luminance and chrominance of the original block in YCbCr color space,  $Y(p), Cb(p), Cr(p)$  are the average values after shutting off subpixels using pattern  $p$ ,  $\gamma_Y, \gamma_{Cb}, \gamma_{Cr}$  are the JND thresholds for  $Y, Cb, Cr$  change. Since the number of candidate pattern for a given OLED display is small and the patterns are ordered based on the power-saving level, we use a binary search to solve the optimization with negligible overhead. For example, an OLED phone with  $\sim 500$  PPI requires a block size  $n = 4$ , which results in 9 patterns based on the pattern generating algorithm and an average number of trials at  $\log_2 9 = 3.2$ . We will evaluate the computation time and power overhead in Section 6.

## 5 IMPLEMENTATION

Although more than 300 PPI is not needed for regular viewing, there may exist a few Apps demanding high pixel density, e.g., VR-related Apps. Therefore, it is inappropriate to enforce subpixel shutoff into all Apps. We implement ShutPix as an application add-on, which captures the rendered content, modifies it, and then sends it to

the display. Developers can simply call ShutPix as a library when drawing the content via `Canvas` or `OpenGL ES`.

At the individual module level, ShutPix first obtains the OLED display resolution and density via `getDisplayMetrics` API. It will then be able to derive the shutoff spatial frequency (block size  $n \times n$ ) via (3) by assuming a smaller viewing distance  $d = 12$  inches [4] and to access the set of candidate shutoff patterns that are generated offline. The default JND thresholds of luminance and chrominance change are set as  $\gamma_Y = \gamma_{Cb} = \gamma_{Cr} = 8$ . We emphasize that the JND thresholds are user-tunable parameters, which can provide flexibility for users with different visual acuity. We will verify the appropriateness of the thresholds in Section 6.

To speed up the image processing, we utilize `RenderScript` instead of pure Java for pixel scanning and processing. We create one reduction kernel and one mapping kernel to carry out the pixel averaging and pattern testing, respectively. These kernels are run with `forEach` function to parallelize individual pixel processing.

## 6 EVALUATION

In this section, we present extensive experiments to evaluate ShutPix under real-world and controlled environment. We also compare ShutPix with existing color transform schemes to demonstrate its advantages as well as its complementary benefits. The experiments are carried out on two OLED smartphones, Motorola Nexus 6 (default device, 493 PPI) and Huawei Nexus 6P (518 PPI).

**Performance metrics.** We measure the display power and system power of the smartphone using the same measurement setup in Section 3.1. In addition, we evaluate the display quality of ShutPix through illustrative results, user studies, and *Global Contrast Loss (GCL)*. GCL is defined as the difference of global contrast, i.e., standard deviation of all pixel values, between two displays. It is a common image quality metric [25] and has been used in display evaluations [2]. Positive GCL indicates contrast loss while negative GCL implies increased contrast. According to [2], a GCL value within  $[-4, 4]$  typically ensures a reasonable display.

### 6.1 Real-world Evaluations

We first perform real-world evaluations when using Android Apps with ShutPix. We embed ShutPix into 10 open-source Apps, which are deliberately selected in the sense that they span through different categories and they are diverse in the background luminance and content/layout complexity. For repeatable experiments, we perform a set of predefined 3-minute actions on each App.

**Power reduction.** Table 1 reports the power reduction when enabling ShutPix. We can see that ShutPix can bring about an average of 20% display power saving. According to the power trace, the system power reduction reaches over 14% on average. In general, applications with complex and diverse media content, e.g., video player or podcast with text, image, audio, etc., achieve more power reduction whereas bright and simple Apps, e.g., PDF reader, present a conservative power saving. This is because power performance is content-dependent. The static Apps only have limited content. If such content is inefficient in terms of subpixel shutoff, e.g., smooth surface, the average power reduction will be compromised.

To highlight the benefits of luminance masking and contrast masking based weighting, we repeat the experiment when all display regions are treated equally, i.e., no weighting. Table 1 shows

**Table 1: Power saving of ShutPix on Apps**

App	Default Setting	No Weighting
ExoPlayer (video)	26%	20%
QKSMS (messaging)	20%	15%
Amaze (file manager)	14%	12%
MuPDF (PDF)	12%	9%
Simple Gallery (photo)	23%	16%
Alarm Klock (clock)	27%	21%
Forecastie (Weather)	16%	11%
RunnerUp (fitness)	27%	20%
AntennaPod (podcast)	21%	16%
spaRSS (news)	17%	14%

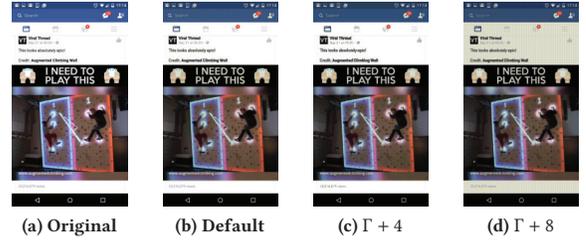
that the average display power saving is dropped to 15%. In this case, ShutPix cannot exploit the visual redundancy effects to shut off as many subpixels as possible. Therefore, we must enable the weighting to maximize the power reduction.

**Illustrative results.** To visualize the impacts on display quality, we show an example screenshot with/without ShutPix and the corresponding measurement results in Figure 10. In casual viewing, we can achieve an extremely similar, if not identical, perception on the default ShutPix content and the original content. This is also consistent with the low GCL value. Indeed, we have conservatively and experimentally selected the default JND thresholds ( $\Gamma = [\gamma_Y, \gamma_{Cb}, \gamma_{Cr}]$ ) to cater to the diverse users and applications. In contrast, when increasing the JND thresholds, users can trade color change for power saving at their own choices. In addition to varying the components of  $\Gamma$  identically, users may also tune  $\gamma_Y, \gamma_{Cb}, \gamma_{Cr}$  individually for fine-grained color change. This indicates the flexibility of ShutPix by employing  $\Gamma$  as a user knob.

It is also interesting to point out that if one zooms in the screenshots of Figure 10b-10d, she should see the overspread spots of subpixel shutoff which are the fingerprint of ShutPix operation. Note that this zoom-in effect will not occur when the user is interacting an App on the fly. In fact, once a pinch-out gesture is detected, the App will generate the zoom-in content in real time and ShutPix is then able to override the zoom-in content just as any other content. Even though the content is magnified, the subpixel shutoff is still kept at the indistinguishable size of physical pixels.

**User study.** To further confirm the display quality, we perform a user study with the 25 users recruited in Section 3.1. We ask them to freely use the 10 Apps as in daily life and each App usage lasts for 3 minutes, where ShutPix is activated either in the first 1.5 minutes or the last 1.5 minutes. The users need to identify whether or not they perceive display quality change. As shown in Figure 11, only 15 out of 250 cases report the perception change. It is interesting to observe that all cases with perception change are found in the PDF, news or messaging Apps, where people may pay more attention to the screen and stay more time viewing a given content.

**Overhead.** The time overhead of ShutPix primarily stems from computing the weights, searching the shutoff pattern, and overriding the content. For other tasks, spatial frequency setup is a simple calculation whereas candidate patterns are pre-derived offline. We measure the execution time of ShutPix and observe an average of 12 ms on different Apps. Such a time overhead will not impact the usage of mobile Apps with regular interaction frequency. Even when using ExoPlayer for playing video, the ShutPix-processed playback is smooth. Another note is that ShutPix is inherently efficient since it only involves binary choice for each subpixel rather than searching  $2^8$  possible colors in traditional color transform.



**Figure 10: ShutPix default setting saves 16% power without degrading viewing experience (GCL:-2.5). The power saving and GCL of (c)(d) are -4.2/5.2 and 19%/34%, respectively.**

Furthermore, we need to ensure that the power overhead of ShutPix is acceptable in order to justify the power saving. We run the Apps with and without ShutPix, and measure the system power. We force ShutPix to show the unchanged content, i.e., no subpixel shutoff, in order to guarantee the same display power in the two cases. The results present an  $\sim 1\%$  power overhead of ShutPix across different Apps. As we have demonstrated in the power reduction results, such a power overhead is very small and does not compromise the performance gains.

## 6.2 Controlled Experiments

We proceed by controlled experiments to study the impacts of different practical factors on ShutPix.

**Impacts of content.** Since the display content in the test Apps may not be extensive, we evaluate ShutPix on a larger dataset with 450 UI and natural images. The dataset covers various Apps, including news, music, email, etc., and numerous image categories, including natural scene, objects, people, etc. We combine *Simple Gallery* with ShutPix to display these content in full screen. We show the cumulative distribution function (CDF) of display power saving by ShutPix in Figure 13 (left). We observe that the power reduction of Nexus 6 ranges from 14% to 25%, within which more than 80% of the content achieve a power saving larger than 18%. The corresponding system power reduction reaches 15% on average. This confirms the applicability of ShutPix to diverse content.

We pair individual power measurement with the corresponding content and discover that darker content usually achieves more power reduction. In addition to the region weighting that prioritizes the subpixel shutoff of such content, another inherent reason stems from their lower average pixel values. Shutting off a dark subpixel (e.g., 30  $\rightarrow$  0) leads to less change of the average pixel value than a bright subpixel (e.g., 250  $\rightarrow$  0), which allows the darker region for more subpixel shutoff and more power saving. Furthermore, although white backgrounds may obtain a less-than-one luminance weight  $W_{lum}$ , they are also plain surface without texture, rendering them a high contrast weight  $W_{con}$ . Thus, they are usually not prioritized for subpixel shutoff.

To quantify the display quality in a large scale, we conduct a duplicate run of experiments on the dataset to obtain GCL. From the results of GCL in Figure 13 (right), we see that more than 92% of the content on Nexus 6 achieve a GCL in  $[-4, 4]$ , which corresponds to satisfactory display quality. This validates the general acceptability of ShutPix in display quality.

**Impacts of device.** We also repeat the above measurement of Motorola Nexus 6 on Huawei Nexus 6P. As shown in Figure 13,

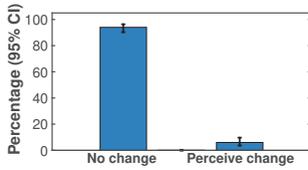


Figure 11: User study on ShutPix perception.

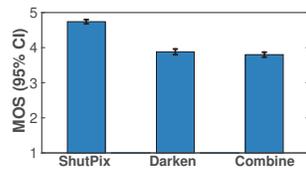


Figure 12: User study on quality comparison.

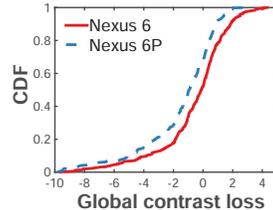
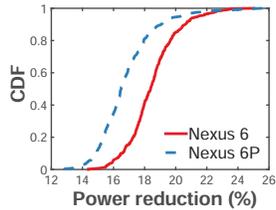


Figure 13: Power saving and GCL over 450 images.

the power saving on Nexus 6P is comparable with Nexus 6. We report that the power saving of Nexus 6P ranges from 13%~26%, which corresponds to an average of 16% reduction in system power. Although existing devices may have distinct display, their OLED power models all follow a power-law function. Therefore, as long as the PPI of an OLED is greater than normal, we expect ShutPix to achieve desirable power reduction. Furthermore, the GCL of Nexus 6P is also desirable. Based on these results, we expect that ShutPix can be applied in more Android smartphones.

### 6.3 Comparison with Color Transform

We now conduct a performance comparison and discuss where ShutPix stands among prior art. Existing color transform schemes include distortion based and user preference based schemes. User preference based schemes sacrifice the content fidelity (e.g., half screen dimmed half screen bright [35]) and only target specific use cases (e.g., web browser [10] and top-down reading [35]). It is thus difficult to directly compare them with ShutPix. We choose distortion based schemes for comparison since they focus on fidelity rather than usability. The benchmark scheme dynamically darkens the OLED display under a distortion threshold (denoted by *Darken*), which represents a large body of existing works, e.g., [31, 32, 39]. We implement this strategy by enforcing the constraint of SSIM since SSIM was used in OLED display [18].

For fair comparison, we aim at comparing the display quality of different schemes under the same power. For each App, we perform the predefined actions and measure the display power when enabling ShutPix. We then repeat this test with Darken and adjust the SSIM threshold such that the display power under Darken matches that under ShutPix. We carry out a user study with the 25 users to assess the quality of ShutPix and threshold-adjusted Darken, as well as a combined scheme that implements ShutPix on top of Darken (denoted by *Combine*). We follow the ITU single-stimulus protocol [22]. The users perform a 1-min predefined interaction on an App under three power-saving schemes and rate the corresponding viewing perception. This is then repeated for all 10 Apps. Although ShutPix and threshold-adjusted Darken achieve approximately the same energy, Figure 12 shows that ShutPix presents the highest mean opinion score (MOS) while Darken clearly incurs perception change. More importantly, Combine has a similar perception as

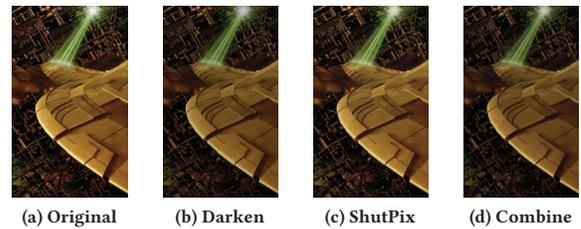


Figure 14: Example screenshot: Under 18.1% and 17.9% power saving, (c) has the best quality while (b) is darker. Although (d) is similar as (b), its power saving is larger (28.8%).

Darken. Given that Combine will definitely save more energy than Darken, this demonstrates the potential of ShutPix as a complementary design over existing color transform schemes. We show an example indicating the above advantages of ShutPix in Figure 14.

## 7 DISCUSSION

**Applicability and system-level implementation.** Besides VR Apps that enforce a shorter-than-normal screen-to-eye distance, ShutPix can save display energy for most Apps in everyday viewing. Although we did not evaluate games, ShutPix shall work on them since the time overhead of ShutPix is small enough to render a normal 60-frame-per-second game. Indeed, the time efficiency may need further improvement to support some higher-frame-rate games [16]. ShutPix is currently implemented at App level and is not yet ready for system UI, e.g., when one swipes the screen in the main menu. We can embed ShutPix in `SurfaceFlinger` to re-composite each frame before display. The ultimate goal is to enable ShutPix in system UI and selected Apps through users' Android *Settings*, which is expected to achieve even more energy saving.

**Viewing distance and visual acuity.** While generally effective, ShutPix might impact the viewing perception of users who have shorter viewing distance and better visual acuity. In this case, the users can simply fine-tune the JND thresholds to alter the subpixel shutoff to a level that is most appropriate for their daily use. One alternative for unusual viewing distance is to use online viewing distance detection. However, current camera or ultrasonic based approaches need complex or even separate hardware [13]. How to integrate these features needs a full-scale study in the future.

## 8 CONCLUSION

In this paper, we have presented a novel framework that achieves systematic power reduction for everyday viewing on OLED smartphones. Inspired by the motivational studies, we propose a suite of designs to optimally shut off the redundant subpixels of display content. Extensive real-world experiments in various practical conditions show that ShutPix can save 21% of display power and 15% of system power without degrading viewing experience. ShutPix represents a complementary direction to existing color transform schemes that only fine-tune every subpixel but not shut them off. The success of ShutPix shall call for combined subpixel shutoff and color transform to further enhance the display energy efficiency.

## 9 ACKNOWLEDGEMENTS

This work is supported by NSF Grant ECCS-1405594.

## REFERENCES

- [1] B. Anand, L. Kecen, and A. L. Ananda. 2014. PARVAI-HVS aware adaptive display power management for mobile games. In *International Conference on Mobile Computing and Ubiquitous Networking (ICMU)*.
- [2] B. Anand, K. Thirugnanam, J. Sebastian, P. G. Kannan, A. L. Ananda, M. C. Chan, and R. K. Balan. 2011. Adaptive display power management for mobile games. In *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*.
- [3] AndroidPIT. Screen Resolution: when is enough, enough? <https://www.androidpit.com/screen-resolutions-when-is-enough-enough>.
- [4] Y. Bababekova, M. Rosenfield, J.E. Hue, and R.R. Huang. 2011. Font size and viewing distance of handheld smart phones. *Optometry & Vision Science* 88 (July 2011), 795–797.
- [5] A. Carroll and G. Heiser. 2010. An analysis of power consumption in a smartphone. In *USENIX annual technical conference (USENIXATC)*.
- [6] H. Chen, J. Wang, W. Chen, H. Qu, and W. Chen. 2014. An image-space energy-saving visualization scheme for OLED displays. *Computers & Graphics* 38 (Feb. 2014), 61–68.
- [7] X. Chen, Y. Chen, Z. Ma, and F. C. Fernandes. 2013. How is energy consumed in smartphone display applications?. In *ACM Workshop on Mobile Computing Systems and Applications (HotMobile)*.
- [8] X. Chen, Y. Chen, and C. J. Xue. 2015. DaTuM: dynamic tone mapping technique for OLED display power saving based on video classification. In *ACM/EDAC/IEEE Design Automation Conference (DAC)*.
- [9] Y.-J. Chin and T. Berger. 1999. A software-only videocodec using pixelwise conditional differential replenishment and perceptual enhancements. *IEEE Trans. Circuits Syst. Video Technol.* 9 (April 1999), 438–450.
- [10] M. Dong and L. Zhong. 2011. Chameleon: a color-adaptive web browser for mobile OLED displays. In *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*.
- [11] M. Dong and L. Zhong. 2012. Power Modeling and Optimization for OLED Displays. *IEEE Trans. Mobile Comput.* 11 (Sept. 2012), 1587–1599.
- [12] L. Fang, O. C. Au, and N. M. Cheung. 2013. Subpixel rendering: from font rendering to image subsampling [Applications Corner]. *IEEE Signal Process. Mag.* 30 (May 2013), 177–189.
- [13] S. He, Y. Liu, and H. Zhou. 2015. Optimizing Smartphone Power Consumption through Dynamic Resolution Scaling. In *ACM MobiCom*.
- [14] S. Joe and F. Y. Kuo. 2003. Remark on algorithm 659: Implementing Sobol's quasirandom sequence generator. *ACM Trans. Math. Software* 29 (March 2003), 49–57.
- [15] C. K. Kang, C. H. Lin, and P. C. Hsiu. 2015. A win-win camera: Quality-enhanced power-saving images on mobile OLED displays. In *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*.
- [16] K. Lee, D. Chu, E. Cuervo, J. Kopf, Y. Degtyarev, S. Grizan, A. Wolman, and J. Flinn. 2015. Outatime: Using speculation to enable low-latency continuous interaction for mobile cloud gaming. In *ACM International Conference on Mobile Systems, Applications, and Services (MobiSys)*.
- [17] D. Li and W. G. J. Halfond A. H. Tran. 2016. Making Web Applications More Energy Efficient for OLED Smartphones. In *European Conference on Computer Systems (EuroSys)*.
- [18] C.-H. Lin, C.-K. Kang, and P.-C. Hsiu. 2014. Catch your attention: quality-retaining power saving on mobile OLED displays. In *ACM/EDAC/IEEE Design Automation Conference (DAC)*.
- [19] Y. Liu, M. Xiao, M. Zhang, X. Li, M. Dong, Z. Ma, Z. Li, and S. Chen. 2016. GoCAD: GPU-Assisted Online Content-Adaptive Display Power Saving for Mobile Devices in Internet Streaming. In *International World Wide Web Conference (WWW)*.
- [20] M. Livingstone and D. H. Hubel. 2002. *Vision and art: The biology of seeing*. Harry N. Abrams, New York, USA.
- [21] E. Messina. 2006. Standards for visual acuity. In *National Institute for Standards and Technology*.
- [22] ITU methodology for the subjective assessment of the quality of television pictures. 2012. ITU-R BT.500-13. (2012).
- [23] M. J. Morgan and R. J. Watt. 1982. Mechanisms of interpolation in human spatial vision. *Nature* 299 (Oct. 1982), 553–555.
- [24] NDTV. Samsung Unveils 5.5-Inch 4K Smartphone Display Prototype for VR. <http://gadgets.ndtv.com/mobiles/news/samsung-unveils-55-inch-4k-smartphone-display-prototype-for-vr-843266>.
- [25] M. Pedersen and J. Y. Hardeberg. 2012. Full-reference image quality metrics: Classification and evaluation. *Foundations and Trends® in Computer Graphics and Vision* 7 (Jan. 2012), 1–80.
- [26] phoneArena. Quad HD vs 1080p display resolution: can people actually see the difference? <http://goo.gl/JVtwBd>.
- [27] phoneArena. Survey shows battery life to be the single main gripe of today's mobile phone user. <http://goo.gl/UrOcxY>.
- [28] Android Open Source Project. Graphics architecture. <https://source.android.com/devices/graphics/architecture.html>.
- [29] Qualcomm. Devices That Report Accurate Battery Power. <http://goo.gl/ZecX96>.
- [30] M. Rabbani and R. Joshi. 2002. An overview of the JPEG 2000 still image compression standard. *Signal processing: Image communication* 17 (Jan. 2002), 3–48.
- [31] P. Ranganathan, E. Geelhoed, M. Manahan, and K. Nicholas. 2006. Energy-aware user interfaces and energy-adaptive displays. *IEEE Computer* 39 (March 2006), 31–38.
- [32] D. Shin, Y. Kim, N. Chang, and M. Pedram. 2013. Dynamic driver supply voltage scaling for organic light emitting diode displays. *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.* 32 (July 2013), 1017–1030.
- [33] J. Spindler, T. Hatwar, M. Miller, A. Arnold, M. Murdoch, P. Kane, J. Ludwicki, P. Alessi, and S. Slyke. 2006. System considerations for RGBW OLED displays. *Journal of the Society for Information Display* 14 (Jan. 2006), 37–48.
- [34] P. Stanley-Marbell, V. Estellers, and M. Rinard. 2016. Crayon: saving power through shape and color approximation on next-generation displays. In *European Conference on Computer Systems (EuroSys)*.
- [35] K. W. Tan, T. Okoshi, A. Misra, and R. K. Balan. 2013. FOCUS: a usable & effective approach to OLED display power management. In *ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*.
- [36] Z. Wei and K. N. Ngan. 2009. Spatio-temporal just noticeable distortion profile for grey scale image/video in DCT domain. *IEEE Trans. Circuits Syst. Video Technol.* 19 (March 2009), 337–346.
- [37] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. 2003. Overview of the H.264/AVC video coding standard. *IEEE Trans. Circuits Syst. Video Technol.* 13 (July 2003), 560–576.
- [38] J. Xue and C. W. Chen. 2012. Mobile JND: Environment adapted perceptual model and mobile video quality enhancement. In *ACM Multimedia Systems Conference (MMSys)*.
- [39] Z. Yan and C. W. Chen. 2016. RnB: rate and brightness adaptation for rate-distortion-energy tradeoff in HTTP adaptive streaming over mobile devices. In *ACM International Conference on Mobile Computing and Networking (MobiCom)*.
- [40] Z. Yan, Q. Liu, T. Zhang, and C. W. Chen. 2015. Exploring QoE for Power Efficiency: A Field Study on Mobile Videos with LCD Displays. In *ACM International Conference on Multimedia (MM)*.
- [41] X. Zhang, W. Lin, and P. Xue. 2008. Just-noticeable difference estimation with pixels in images. *Journal of Visual Communication and Image Representation* 19 (Jan. 2008), 30–41.