CeilingCast: Energy Efficient and Location-Bound Broadcast Through LED-Camera Communication

Jie Hao* Yanbing Yang* Jun Luo
School of Computer Engineering
Nanyang Technological University, Singapore
Email: {haojie, yyang017, junluo}@ntu.edu.sg

***Abstract*—Although Visible Light Communication (VLC) is gaining increasing attentions in research, developing a practical VLC system to harness its immediate benefits using Commercial Off-The-Shelf (COTS) devices is still an open issue. To this end, we develop and deploy CeilingCast as a location-bound wireless broadcast system using COTS LEDs as transmitters and smartphone cameras as receivers. CeilingCast innovates in its effective coding and efficient decoding schemes, so that it can be fully hosted in a smartphone and is feasible for all possible indoor environments. Moreover, we analyze the impact of various parameters on the performance of CeilingCast, in order to derive a model for such VLC systems enabled by COTS devices and hence provide general guidance for future VLC deployments in larger scales. Finally, we conduct extensive field experiments to validate the effectiveness of our LED-camera VLC model, as well as to demonstrate the promising performance of CeilingCast under various parameters.**

I. Introduction

Visible Light Communication (VLC), using visible light spectrum as the communication media, has a long history that can be traced back to 19th century. Although it has been holding the promise of communications with extremely high data-rate and very low-cost for a few decades, it is the boosting market of Light Emitting Diode (LED) that has drastically accelerated the development of VLC as a supplement to existing wireless standards (e.g., Wi-Fi). As LEDs are becoming the pervasive lighting source for indoor environment such as shopping malls and office/residence buildings, and they have a ready access to power/information networks, we now have a handy infrastructure for implementing VLC-enabled wireless systems. Nevertheless, while major research efforts have been made to boost the capacity of VLC [1], [2] and to enable indoor localization [3], [4], [5], harnessing the communication ability of LED-VLC using Commercial Off-The-Shelf (COTS) devices seems to be largely neglected.

Following the seminal work of [6], implementing a receiver applying the rolling-shutter effect of a CMOS camera has become a de facto standard for VLC using COTS devices [4], [7]: only a customized light sensor may serve as an alternative because light sensors in smartphones are optimized for dynamic range rather than response speed [8]. So it appears to be a common belief that enabling VLC using COTS devices is an addressed problem. However, the system built in [6] requires a secondary medium, a plain surface (to avoid interference from textures), for light reflecting. While the Signal Noise Ratio (SNR) of such reflected VLC could be very low in a well-lit room, the difficulty in finding a satisfactory surface in an indoor space may substantially confine the practicality of such a system. Consequently, the validity of the derived models and insights also become questionable.

In fact, VLC has quite a few advantages over existing wireless infrastructures (e.g., Wi-Fi and BLE) even without its high data-rate promise. First of all, the communication infrastructure is virtually free due to the default lighting requirement of indoor spaces. Secondly, the energy consumption of LED transmission is negligibly low as it is incurred only by the control units. Last but not least, the location-bound communication ability of VLC meets perfectly the need from the long envisioned location-based service: coupons or advertisements can be delivered exactly according to the users' locations. Practical VLC implementations using COTS devices may immediately bring these benefits to us, while allowing for a better understanding of future VLC deployments from theoretical and modeling perspectives.

In this paper, we re-visit the issue of realizing VLC-based broadcast using COTS devices, in particular common LEDs as transmitter and smartphone cameras as receiver. On one hand, we aim to build a practical system that works perfectly under normal lighting conditions. We design a coding mechanism that enables cooperative transmission among multiple close-by light sources, and we also innovate in an efficient decoding scheme that suits smartphone receivers. On the other hand, we intend to extract model parameters (e.g., which factors affect the data rate) and hence provide guidelines on future deployments of VLC system. Our major contributions are:

- We build CeilingCast as a practical wireless broadcast system, using common LEDs as transmitters and smartphone cameras as receivers, without relying on media other than the light itself.
- We are the first to employ rateless codes to enhance reliability and also to improve the network throughput through cooperative transmissions.
- We also innovate in a lightweight decoding scheme that avoids intensive image processing and is hence amenable to a full smartphone implementation.
- We develop a communication model to characterize this LED-camera VLC by extracting parameters from the

* Equal contributing authors. This work was supported in part by AcRF Tier 1 Grant RGC5/13 through NTU Complexity Institute.

experience in designing and deploying CeilingCast.

- We conduct extensive field experiments to validate the effectiveness of our LED-camera VLC model and to demonstrate the promising performance of CeilingCast.

The rest of our paper is organized as follows. We first survey related literature in Sec. II. Then we illustrate the system architecture of CeilingCast in Sec. III and analyze the factors impacting system performance in Sec. IV. The extensive evaluations on CeilingCast are reported in Sec. V. Finally Sec. VI concludes this paper.

II. Related Work

VLC has been favored as a complementary approach to traditional wireless communications over the last decade, as it has the potential to offer broadband communication on unlicensed spectrum with a high degree of space reuse. The recent standard 802.15.7 [9] has defined specifications while categorizing VLC into two classes: high rate and low rate. High rate VLC can achieve up to gigabits per second with specialized high speed photodiode receivers, whereas low rate VLC is generally explored on COTS mobile devices. As we focus only on the latter in this paper, so are the following discussions on the related literature.

Screen-Camera Communication is one important branch of low rate VLC. It mainly focuses on designing sophisticated coded images to boost data rate [10], [11], [12], [13], or to enhance link reliability [14], [15], [16]. To improve data rate, SBVLCD [10] and COBRA [11] exploit sophisticated barcode design, PixNet [12] leverages efficient modulation mechanism, and SoftLight [13] employs channel coding. Reliability is addressed in terms of either frame synchronization [16] or barcode detection under a prolonged communication distance and device diversity [14], [15]. Moreover, recent studies intend to enhance viewing experience by hiding information in a given screen content without interfering communications. PiCode [17] integrates barcodes with existing images to reduce the visual artifacts, while HiLight [18] avoids modifications on RGB values by utilizing the alpha channel to encode bits into the pixel translucency changes.

LED-Camera Communication mostly exploits the rolling-shutter effect of CMOS camera on smartphones. Unlike the high rate VLC where similar LED transmitters are used, the link capacity in this case is confined by the rolling-shutter features and the smartphone capability, which is where innovations can be made. Recent research [6] achieves data rates up to 3.1kBps at 20fps. However, it requires a second media, i.e. a plain surface, to reflect lights so as for the camera to capture the banded images, making its practicality questionable. As an important application of this type of VLC, Visible Light Positioning (VLP) [5] makes use of the limited bandwidth to transmit location identifers. Moreover, Luxapose [5], for the first time, proposes to use light as the only communication media, and this is further extended by RollingLight [19] to build a new LED-camera VLC system. Nevertheless, both proposals require heavy decoding computations that may not be feasible for COTS devices, and they treat several transmissions independently rather than leveraging them for throughput/reliablity enhancement. Our CeilingCast is designed to fill these gaps so as to make VLC practical, and it achieves a throughput far higher than RollingLight.

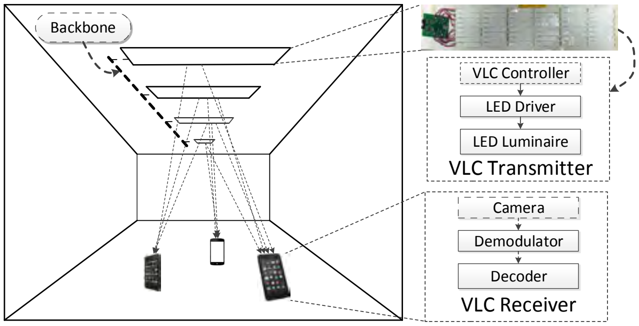


Fig. 1. System architecture of CeilingCast.

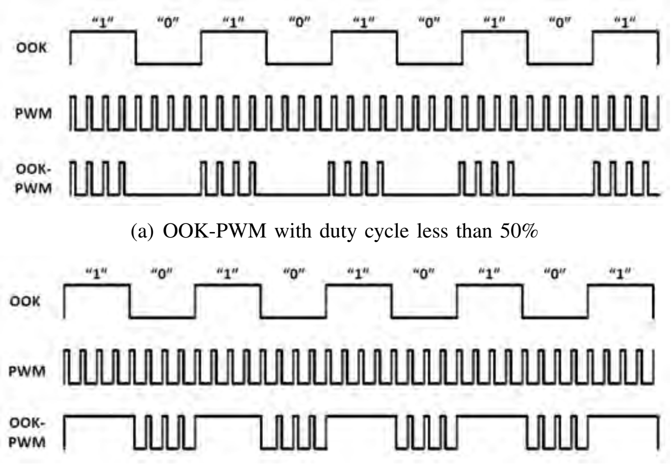
III. System Architecture

A practical LED-Camera VLC system consists of two main components: COTS LED luminaires as stationary transmitters to emit intensity modulated light and smartphones as receivers. A user holding a smartphone may freely wander in the indoor environment and receive location-bound information from nearby LED luminaires. Fig. 1 shows the system overview of CeilingCast; it exploits the rolling-shutter effect of CMOS cameras to decode the modulated signals. As a rolling-shutter exposes an photo in a column-by-column manner, it leads to a banded image when shooting an intensity modulated luminaire. This image carries the information sent by the luminaire in a way that bright bands indicate ON while dark bands indicate OFF. In this way, we may receive intensity modulated signals up to several kHz given that the scanning rate of a rolling-shutter can often match this frequency and produce bands with measurable widths. More detailed (graphical) explanations can be found in [4], [19].

In this section, we focus on presenting our innovations in designing the coding/decoding mechanisms. In particular, we explain how we may combine several transmissions (from multiple close-by luminaires) to achieve higher throughput and reliability. And we also describe an efficient decoding scheme that avoids computation intensive image processing.

A. A Hybrid OOK-PWM Modulation

CeilingCast employs a hybrid modulation of On-Off Keying (OOK) and Pulse-Width Modulation (PWM) as shown in Fig. 2. As the simplest modulation method, OOK is universally employed on VLC, where the LED is switched ON or OFF to represent data bits “1” or “0”, respectively. We further embed a high frequency PWM in each OOK bit for dimming control: it allows the light intensity of an LED luminaire to be adjusted without either causing flickering or interfering data transmission. We refer to Sec. III-E for more details. As the PWM is supposed to be undetectable by the rolling-shutter effect at the receiver, it runs at 80kHz in CeilingCast.



(a) OOK-PWM with duty cycle less than 50%

(b) OOK-PWM with duty cycle more than 50%

Fig. 2. OOK-PWM modulation: the combination of OOK-PWM effectively avoids flickering and also allows for dimming control.

The packet structure in CeilingCast is set as shown in Fig. 3. Each packet has a preamble of bit string "11110" and it always ends with a single bit "0". The Packet Sequence Number (PSN) comes after the preamble and it is followed by the payload. For bits in both the PSN and payload, CeilingCast applies Manchester coding [6] for its simplicity and absent of DC component. As Manchester coding guarantees no more than two adjacent "1" or "0" OOK bits, the preamble generates the widest bright band in the image and is hence easy to be detected as the indication of the start of a packet.

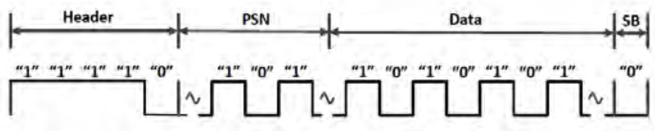


Fig. 3. CeilingCast packet structure: a packet consists of a preamble "11110", up to 8-bit PSN, data payload, and a tailer "0".

B. Encoding and Code Assignment

As LED-Camera VLC is normally unidirectional, a transmitter has no knowledge about the reception status of a receiver. Therefore, a Forward Error Correction (FEC) scheme has to be in place to combat the packet loss. Moreover, simultaneous transmissions from multiple luminaires also call for an encoding mechanism that enables the transmitters to cooperatively improve the throughput and reliability. To this end, rateless codes appear to be an appealing choice, as it encodes k original packets into potentially infinite number of packets so that the receiver can successfully recover the original packets by receiving any $m > k$ encoded packets.

Among all rateless codes, Raptor code [20] comes with linear time encoding and decoding, causing low computation complexity and decoding overhead. Therefore, CeilingCast adopts Raptor code and its implementation [21]. As shown in Fig. 4, a set of intermediate packets are firstly derived from the original packets so that the intermediate packets can sufficiently reconstruct the original ones. Repair packets are then produced by applying LT encoding; each is derived by XORing a number of intermediate packets. The final encoded packets are the combination of the original and repair ones. As CeilingCast has no acknowledgement from camera for the transmitter to stop coding and sending, we regulate the transmitter to send out only $n = 1.25k$ encoded packets for every k original packets. Upon receiving slightly more k packets, a receiver uses Gaussian elimination to start recovering the original ones.

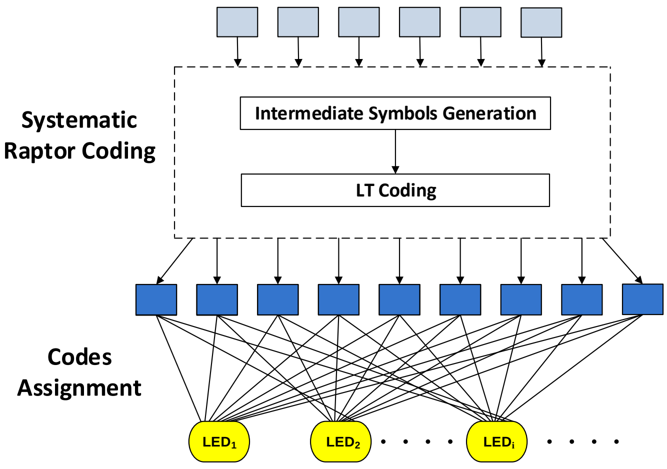


Fig. 4. Raptor coding and codes assignment.

As CeilingCast relies on the lighting system to transmit packets, it has the opportunity to leverage multiple close-by luminaires to perform cooperative transmissions for improving both throughput and reliability. Imagine a simple scenario that a smartphone receives messages from two luminaires. Suppose we let the luminaires to work independently, then either we waste the capacity of one luminaire if the message is sent by only one luminaire or individual packets loss may ruin both messages if two luminaires are transmitting. However, if we allow the two luminaires to cooperatively transmitting, then we may either reduce the transmission time or let the two transmissions complement each other using the Raptor coding. In fact, the reliability enhancement is particularly useful when a user is moving. Nonetheless, the challenge here is how we assign the encoded packets to individual luminaires, so that every received packet from any of luminaire can effectively contribute to decoding.

Suppose there are $N_L : N_L \ll n$ luminaires in a cooperative domain, and they are labeled as $\mathsf{LED}_0, \mathsf{LED}_1, \cdots, \mathsf{LED}_{N_L-1}$. CeilingCast currently adopts a cyclic assignment scheme in which it assigns all the n encoded packets (codes) to each luminaire but in a circular shift manner, as illustrated in the lower part of Fig. 4. In particular, we set the offset of the i-th luminaire as $o_i = i\lceil n/N_L \rceil, 0 \le i \le N_L - 1$, then the sequence numbers of the codes assigned to this luminaire is $\{o_i, o_i + 1, \cdots, n-1, 0, 1, \cdots, o_i - 1\}$. Apparently, this assignment may increase the throughput by N_L times if the reception from each luminaire is perfect, and it certainly allows for quick compensations of lost packets. In a large indoor facility, there can be multiple cooperative domains, and our assignment scheme is applied individually to each of these domains. Since the luminaires are all connected to the same power grid, Power Line Communication (PLC) [22] can be used for scheduling transmissions.

C. Pre-Decoding: RoI Extraction

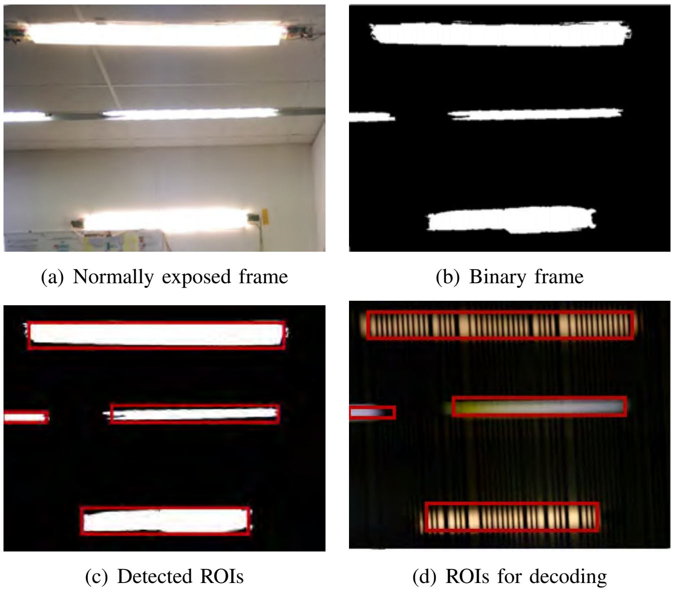
Given a frame captured by a camera, only the banded sections in it contain information to be decoded. Such an information-containing region is termed Region of Interest (RoI) [23]. Though it is visually straightforward to recognize these RoIs, extracting them automatically is far from trivial. Existing proposals [5], [19] resort to sophisticated Computer Vision (CV) techniques to solve the problem, introducing such a high computational overhead that the computations have to be offloaded to a cloudlet server [5].

Our innovative scheme makes use of the default function of the camera to produce RoI masks so that the intensive image processing can be largely avoided. During a continuous reception, the first frame of every p frames is exposed normally (e.g., 1/60s) as shown in Fig. 5(a) and the rest are quickly exposed (e.g., 1/7500s) to obtain Fig. 5(d). The first frame is then converted to a binary RoI mask for extracting the RoIs from the remaining $p-1$ frames. As a normal exposure would overexpose the luminaires, we convert the first frame to a binary one shown in Fig. 5(b) by setting pixels with Y value (the luma component of YCbCr color space) exceeding threshold Y_t to "1" and otherwise to "0". Then the algorithm builds a rectangular contour for each cluster of "1" pixels to create the RoI mask shown in Fig. 5(c). Obviously, the information to be decoded from the remaining frames can only exist within these contours. As the frames are shot within a very short period of time, we assume there are no tilting/shifting between the mark and the remaining information frames. However, we are also on the way to design an gyroscope-assisted algorithm to compensate for hand motions.

To establish a baseline, we also propose a CV-based RoI extraction scheme. Basically, each frame first goes through a Gaussian blur, and then the aforementioned binary conversion (with a different threshold) and contour detection procedures are used to extract the RoIs. All these functions are carried out by using OpenCV for Android, and this scheme is already a much simplified version of those used in [5], [19]. Obviously, the Gaussian blur procedure is both time and energy consuming. In fact, we have to scale down the frame resolution from 2448×3264 to 768×1024 in order for the CV-based scheme to work in a smartphone. As we will show in the evaluation, the frame resolution can greatly affect the throughput.

D. Demodulation and Decoding

The traditional demodulation methods often pre-determine a set of sampling times and then compare the samples with a threshold. However, the blooming effect of a camera (bleeding or smearing photons from saturated pixels to adjacent pixels) brings difficulty in determining the sampling times.


(a) Normally exposed frame

(b) Binary frame

(c) Detected ROIs

(d) ROIs for decoding

Fig. 5. Lightweight RoI extraction: for each normally exposed frame (a), we convert it to a binary frame in (b) and detect contours in (c), so that RoIs in a quickly exposed frame (d) can be extracted.

For CeilingCast, we make demodulation decisions by reasoning on the widths of bright/dark bands. Given the packet structure described in Sec III-A, we can only have four patterns to be decoded after the preamble, namely $W_{1b}\backslash W_{1d}$, W_{1d}/W_{1b}, $W_{2b}\backslash W_{2d}$, and W_{2d}/W_{2b}, whereas the preamble has a signature of W_{4b}, where W_{1b} represents the width of one bright band and likewise for other symbols. As the blooming effect diminishes with the number of consecutive bright bands, we derive 5 relations between the symbol width W the aforementioned quantities: i) $W = W_{1b} + W_{1d}$, ii) $2W = W_{2b} + W_{2d}$, iii) $W_{2b} <= 2W_{1b}$, iv) $W_{2d} >= 2W_{1d}$, and v) $W_{4b} \approx 2W$, where $W = W_r F_r / F$ is determined by the rolling-shutter frequency and scanning (column) width F_r and W_r, respectively, as well as the working frequency F of an LED luminaire.

For each RoI, the first step involves choosing a threshold to distinguish bright and dark bands. Early proposals suggest using polynomial regression [6] to determine the threshold due to low SNR. CeilingCast, with a sufficient level of SNR, achieves a satisfactory detection accuracy by simply setting the threshold as the average of the maximum and minimum pixel illuminance values within the RoI. The next step is to estimate W_{1d} and W_{1b}. It starts by finding two widest bright bands of width W_{4b}, between which there should be fixed numbers of bright and dark bands given the defined packet structure. So W_{1b} and W_{1d} are estimated by averaging total dark bands and bright bands, respectively. Finally, the demodulating procedure starts by using the estimated values and the aforementioned rules for distinguishing among the four possible patterns. This procedure goes on until all the bits are recognized.

After the demodulation, Raptor decoding procedure takes over with the knowledge of the size of a packet and the number of original packets k. Upon receiving $m = k(1 + \epsilon)$

encoded packets ($\epsilon = 0.15$ is chosen to be the overhead of LT codes for CeilingCast), Gaussian elimination is used for decoding. As one frame may capture multiple RoIs, the decoder waits for all RoIs to be demodulated and combines all received packets in one frame for decoding, so as to reduce the decoding latency dramatically. As mentioned in Sec. III-B, this allows for almost immediate compensation of packet loss and a manyfold increase in throughput.

E. Dimming

Dimming is an important requirement posed by IEEE 802.15.7. As mentioned in III-A, CeilingCast can support wide range dimming duty cycle by its hybrid modulation. As the duty cycle of OOK D_{OOK} is fixed given the defined packet structure, we can tune that of PWM D_{PWM} to meet a certain dimming requirement. Specifically, we can tune the overall duty cycle D from 0 to D_{OOK} by embedding the PWM signal into OOK "1":

$$D = D_{\text{OOK}} \times D_{\text{PWM}}, \tag{1}$$

as shown in Fig. 2(a). To reach a full range of D, we can further embed the PWM signal into OOK "0" so that the value of D can go all the way up to 100%:

$$D = D_{\text{OOK}} + (1 - D_{\text{OOK}}) \times D_{\text{PWM}}, \tag{2}$$

as shown in Fig. 2(b). Obviously, making D too close to 0 or 100% can hurt the SNR and thus the throughput, so a preferable range of D for CeilingCast is from 10% to 70%.

IV. Performance Analysis

In this section, we analyze the performance-impacting factors and present the achievable performance of CeilingCast under realistic constraints.

A. Receiver-Camera Configurations

The major issue hampers the performance of CeilingCast is the inter-symbol interference caused by the blooming effect. Ideally, bright and dark bands should have the same width and the demodulation should be trivial. However, the blooming effect causes a bright band to leak into a neighboring dark band, resulting in irregular widths in bands. While reducing exposure index may suppress this effect, it affects SNR too. So we intend to verify the impacts of various combinations of LED illuminance and camera sensitivity settings on both the SNR (contrast in case of CeilingCast) and blooming effect.

We have three parameters at hand: illuminance, ISO and exposure time; they control output of each pixel (sensor) by adjusting the incident light intensity, sensor sensitivity, and time to receive photons, respectively. We vary the LED illuminance from 153lux to 301lux (corresponding to those obtained at 1.5m distance with the PWM duty cycle from 10% to 25%), the ISO sensitivity from 100 to 300, and exposure time from $1/7500$s to $1/5000$s, and the exposure index is proportional to the product of illuminance, ISO and the reciprocal of exposure time. In order to unify contrast and blooming effect into the same perspective, we normalize them before plotting the results in Fig. 6. The normalized contract is computed as $(Y_b - Y_d)/Y_{\max}$, with Y_b, Y_d, and $Y_{\max}$ being the luminance values of the bright band, the dark band, and the camera specified maximum. We indicate the blooming effect by W_{1b} and compute its normalized value as $(W_{1b} - 0.5W)/(0.5W)$.

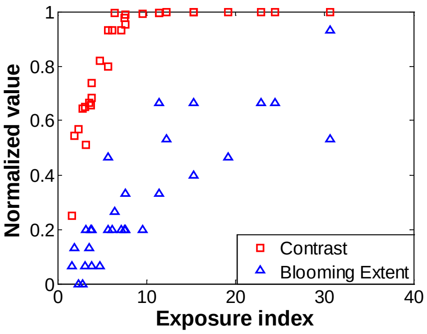


Fig. 6. The image quality with varying exposure indices. A higher exposure index leads to a higher contrast but also a more severe blooming extent.

Fig. 6 shows that, though the contract increases with the exposure index initially, it quickly gets saturated. At the same time, the blooming effect appears to grows linearly with the exposure index. In fact, when we reach the highest exposure index, W_{1b} reaches up to 32pixels, almost occupying its neighboring dark band. Apparently, the optimal exposure index should be right before the saturation of the contrast, where the blooming effect is still well controlled. Considering that the bands can be even narrower if the LED frequency is further increased to enhance throughput, CeilingCast could be more sensitive to the blooming effect. Therefore, given a normal indoor illuminance between 200 to 700lux by standard illumination requirement, we tune the PWM duty cycle to meet the requirement, while choosing ISO as 200 and exposure time as $1/7000$s in the office environment where CeilingCast is deployed and tested; more detailed justifications on these settings will be presented in Sec. V-B.

B. System Model

We hereby explain how we set the packet length packet (in particular the payload length according to the structure specified in Sec. III-B) and also derive the achievable bit rate given several design parameters. We first set up the relations between key system parameters and the admissible packet length (thus bit rate), then we deduce their quantities based on the specific devices adopted by our currently implementations. In addition to the definitions in Sec. III-D, we further assume that i) the LED transmitter has a length L, ii) the camera receiver has a focal length f_c, sensor width S_w, frame width I_w, column resolution P_c, and frame rate R, and iii) the communication distance is less than d and there are N_L transmitters within this range, we have:

- The projected RoI in a frame has the length of $L_{\text{RoI}} = \frac{\phi L}{d} \le P_c$, where $\phi = f_c I_w / S_w$.
- The number of OOK bits an RoI can cover should satisfy $N \le \left\lfloor \frac{L_{\text{RoI}} F}{W_r F_r} \right\rfloor \le \frac{P_c F}{W_r F_r}$.

- To ensure that an RoI can cover at least one complete packet under asynchronous transmission, the packet size in OOK bits should satisfy $P_{\text{size}} = N/2 \leq \left\lfloor \frac{L_{\text{RoI}}F}{2W_rF_r} \right\rfloor = \left\lfloor \frac{\phi LF}{2dW_rF_r} \right\rfloor$.
- According to above analysis, we can bound the OOK bit rate of the CeilingCast as:

$$C = N_L R P_{\text{size}} \leq N_L R \left\lfloor \frac{\phi LF}{2dW_rF_r} \right\rfloor \leq N_L R \frac{P_c F}{2W_rF_r} \qquad (3)$$

Now we can put these parameters into practical perspectives in order to check the actual quantities of packet size and achievable bit rate of CeilingCast. Given the LG Nexus 5 as the CeilingCast receiver, we have $f_c = 30.55$mm, $R = 30$fps by the phone specification, and $F_r = 19$kHz $W_r = 4$pixel through our own measurements. Also, the image sensor has a full resolution 3264×2448 and a preview resolution 1024×768, which allow us to derive $\phi = 3110$pixel under full resolution and $\phi = 976$pixel under preview resolution. As the working frequency of LED can go up to a few GHz, the working frequency of CeilingCast is restricted mainly by the capability of the receiver, namely the rolling-shutter parameters F_r and W_r of the camera.

Given $F_r = 19$kHz, $F \leq F_r/2 = 9.5$kHz has to be met due to the asynchronous nature of the communication. We choose $F = 8$kHz for reliability purpose, which results in a symbol width $W = 19$pixel at full resolution and $W = 6$pixel at preview resolution. Apparently, should we have a higher F_r, the preview resolution would become a bottleneck, as we need some "guard pixels" to combat the blooming effect. In an indoor environment, the ceiling is around 3m from a hand-held smartphone camera sitting right below it. Considering a 0 to 4m horizontal distance, we have $d = 5$m as our maximum communication distance. Substituting $F = 8$kHz and $d = 5$m into earlier formulas, we obtain $N \leq 78$ and thus $P_{\text{size}} = 38$bit. As this is counted as OOK bits and we use Manchester coding, the actual digital bits can be transmitted are 16. Therefore, CeilingCast can offer a bit rate of 480bps with one transmitter at the maximum communication distance, given the frame rate $R = 30$fps of the camera. This bit rate can be further improved by combining more transmitters and using the full resolution of the camera. For example, if we have 6 transmitters within the communication range and we use full resolution for decoding, the bit rate can be boosted to 2880bps in theory.

For a single transmitter, there are other ways to improve the bit rate. According to Eq. 3, reducing the communication distance can increase the bit rate linearly. Also, using a smartphone camera with a higher frame rate can also proportionally increase the bit rate. For example, if we use iPhone 5s with a frame rate 120fps, the bit rate can be tripled.

C. Practical Considerations

The above discussions assume rather ideal scenarios, but various interferences exist in a realistic deployment. One may expect the ambient lighting to be a major interference to CeilingCast communications, but the fact is that ambient lighting only causes noticeable noise if a light sensor is used as the receiver [8] or if the rolling-shutter sensing is applied to some surface reflection [6], [7]. In our case, as CeilingCast directly uses the LED luminaires as the signal source, the SNR is so high that we can run a receiver with an extremely short exposure time, suppressing the interference (ambient light "leaking into" the RoIs) to the largest extent. Therefore, the interference from ambient lighting can surely be neglected for CeilingCast.

One major challenge CeilingCast faces is user motion, including hand micro-motion and mobility during data reception. As we have cooperative transmissions to handle the user mobility during reception, we shall only focus on the interference caused by hand micro-motion. Note that if a user totally moves out of a cooperative domain, the reception will fail but this is indeed the purpose of CeilingCast's location-bound communication: the data service is location dependent.

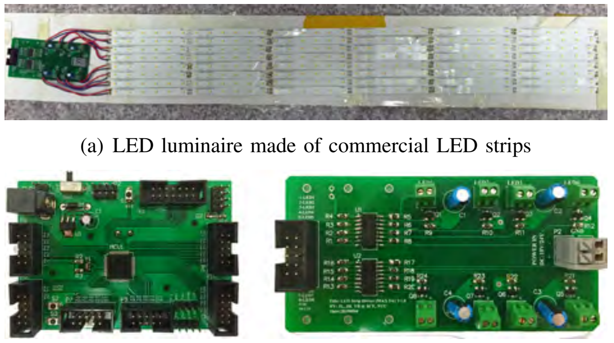
For the hand micro-motion, we assume that its magnitude so minor that it does not cause mis-capturing of the transmitters, which is reasonable for users with normal physical conditions. Therefore, the major problem caused by hand micro-motion is twofold: on one hand, minor camera titling/rotating may distort the RoIs extracted from the first normal exposed frame, causing bit loss within an RoI. On the other hand, major camera tilting may cause certain RoIs un-decodable. While the minor changes can be compensated by gyroscope-assisted geometric transformations, we certainly need to constrain the major ones. Given a tilting angle α, we obtain the length of RoI $\frac{\phi L}{d} \times \cos(\alpha)$. Consequently, the bit rate in Eq. 3 is constrained by $C \leq \cos(\alpha) N_L R \left\lfloor \frac{\phi LF}{2dW_rF_r} \right\rfloor$. In our current implementation, we choose $P_{\text{size}} = 38$bit subject to $N = 78$bits, so the rotation angle should be less than $\arccos(38 \cdot 2/78) \approx 13^\circ$ without incurring bit loss. Fortunately, we can let the application to alert an user for such packet loos and hence rely on user assistance to improve the performance.

V. Evaluation

We report the evaluation results on our CeilingCast testbed in this section. We first explain the experiment settings, and then present the performance evaluations of CeilingCast with respect to various parameters.

A. Experiment Settings

We build CeilingCast transmitters with commercial LED strips [24] and self-developed LED driver. Fig. 7 shows one such transmitter; it includes an LED luminaire, an driver board, and a control unit. The LED luminaire is made of 16 LED strips each containing 36 LED chips, hence a dimension of 120cm×8cm similar to a common fluorescent luminaire. As this LED luminaire has a nominal drive current of 600mA, we use low-cost transistors to build the driver circuit. To enhance the stability and readability of the transmitter, optocouplers are employed between the control unit and driver. In total we deploy 3 such transmitter every 1.5m in parallel in our lab, with one on the wall and the other two on the ceiling, as shown



(a) LED luminaire made of commercial LED strips

(b) VLC control board and LED driver

Fig. 7. CeilingCast transmitter.

in Fig. 8. The one on the wall is deployed for performance evaluation at different transmission distances, while the other two are meant to emulate a realistic setting. We use a LG Nexus 5 smartphone as the receiver, with parameters specified in Sec. IV-B. A message is divided to blocks, each contains k packets. Raptor code [21] encodes at the rate of $n = 1.25 \times k$. Given that the PSN field in a packet contains up to 8 bits, each block can have at most 256 coded packets. The decoding overhead is set $\epsilon = 0.15$. Each packet is transmitted repeatedly within a frame duration 33.3ms.

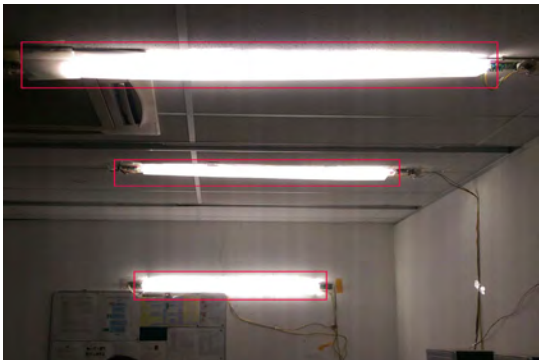
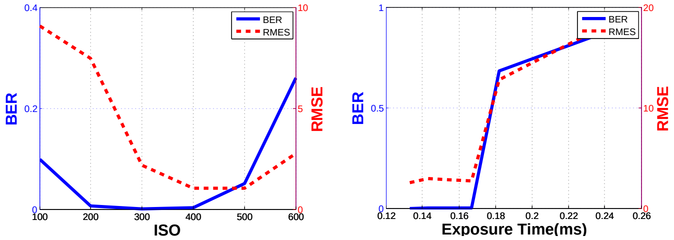


Fig. 8. CeilingCast testbed: three LED luminaires are mounted on wall and ceiling with an interval of 1.5m, and the ceiling is 2.5m above the floor.

B. Camera Settings

In Sec. IV we have analyzed how to choose the exposure time and ISO of the camera to strike a balance between high contrast and low blooming effect. Here we focus on the impact of camera setting on Bit Error Rate (BER) and Packet Error Ratio (PER); both of them take the missing frames into account. In Fig. 9, we use Root Mean Square Error (RMSE) to measure the fluctuation in the widths of the bands. It indicates the difference between the widths of all detected bands and the widths of bands leading to accurate bit decision. RMSE has direct impact on BER because the width of a band serves as the criterion for decoding. In Fig. 9(a), we fix the exposure time at $1/7000$s, the LED duty cycle at 20%, and communication distance at $3m$. With ISO varying from 100 to 600, RMSE and BER first decrease as the contrast is enhanced and then degrade as the blooming effect causes more irregular widths of bands. Fig. 9(b) illustrates how the exposure time affects the performance. We fix the ISO at 200 and other parameters as before. Due to the same reason of blooming effect, RMSE and BER remain stably low at short enough exposure time and then increase when the exposure time exceeds $1/6000s$. The result confirms to our analysis in Sec. IV. Therefore, CeilingCast by default sets ISO at 200, exposure time at $1/7000s$ and duty cycle at 20% unless otherwise specified.



(a) BER vs. ISO

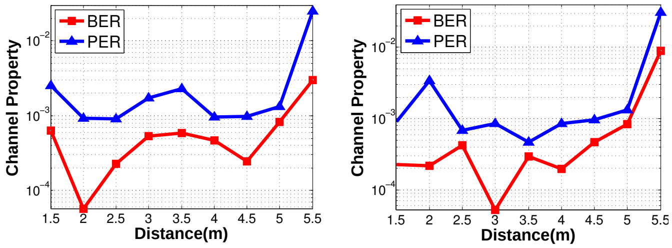
(b) BER vs. Exposure time

Fig. 9. The impact of camera settings. (a) BER reaches the lowest at ISO from 200 to 400. (b) BER increases with the exposure time.

C. Channel Property

Firstly, we evaluate the channel properties with the camera fixed at a stationary position. In Fig. 10 we measure the channel properties in terms of BER and PER at different communication distances from 1.5m to 5.5m, and we also compare the performance of our non-CV-based decoding method with the baseline CV-based decoding by running the receiver with the preview resolution. We observe that both BER and PER stay at a low level regardless of the increasing communication distance until the $d > 5$m. At $d = 5.5$m, BER and PER are dynamically increased because the the projected RoI becomes too small to cover $2P_{\text{size}}$, causing non-negligible decoding failures. Also, the performance in terms of BER and PER of the two decoding methods appear to very similar, proving the superiority of our non-CV-based method: the computation time is about 9ms in our case, whereas the baseline CV-based method takes more than 30ms to complete.

Secondly, we evaluate the channel properties under mobile scenarios. We only use the CV-based decoding here due to drastic changes in the camera position. This experiment



(a) Non-CV-based decoding

(b) CV-based decoding

Fig. 10. Channel properties under stationary scenarios. The channel is stable until the communication distance goes beyond 5m.

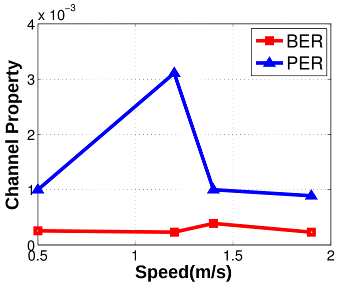
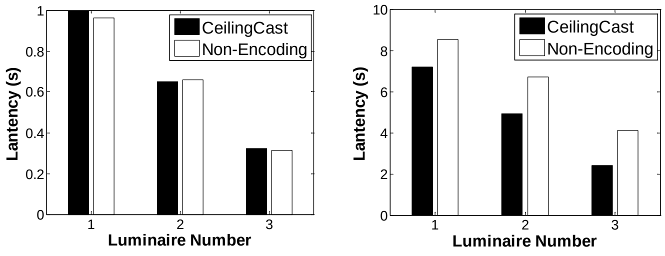


Fig. 11. Channel properties under mobile scenarios. Both BER and PER stay at around 10^{-3} when users moving within a reasonable communication range.

is conducted by two users. They hold the smartphone and walk arbitrarily within the communication range with different speed, while guaranteeing that the camera can always capture the targeted LED luminaire. During the experiment, the accelerometer on the phone is turned on to monitor the average walking speed. Fig. 11 shows that the BER and PER under mobile scenarios are comparable to those of stationary scenarios, confirming the robustness of CeilingCast against mobility. Both these two experiments allow us to conclude that CeilingCast provides stable channels with very low BER/PER ($\sim 10^{-3}$) within a reasonable communication range.

D. Throughput and Latency

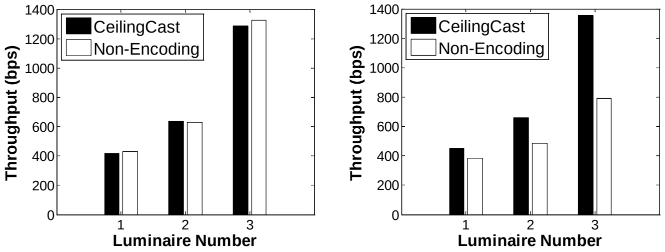
In this section, we evaluate the benefit of using Raptor codes in terms of throughput and latency, by comparing CeilingCast with a baseline non-encoding mechanism that simply sends the set of k original packets assigned to individual lumiaires in a circular shift manner. The latency is the elapsed time from when the camera starts to receive data to when it recovers (for CeilingCast) or receives (for the baseline mechanism) all original packets. In an LED-camera communication session, the latency is determined by the frame rate of the camera since the frames are taken about every 33.3ms that dominates other factors such as decoding computations. Therefore, the latency is approximately proportional to the numbers of camera frames needed for successful decoding. The throughput is computed as the total bits of the k packets divided by the latency. We conduct the experiments with $k = 26$ and $k = 204$ to evaluate how CeilingCast performs under different k values.



(a) k=26

(b) k=204

Fig. 12. Latency vs. luminaire number. The improvement in terms of latency grows with the luminaire number when $k = 204$.



(a) k=26

(b) k=204

Fig. 13. Throughput vs. luminaire number. The improvement in terms of throughput grows with the increasing luminaire number when k=204.

We first evaluate the performance of CeilingCast under stationary scenarios as explained in Sec. V-C. The achieved latency and throughput are plotted in Fig. 12 and Fig. 13, respectively. We can observe that both metrics get improved proportionally to the number of luminaires, and the throughput of CeilingCast can reach 1.35kbps with 3 luminaires. In fact, CeilingCast has the potential to capture at most 6 luminaires in one frame, so it can offer a throughput up to 2.7kbps. We can also observe that, when k is small, CeilingCast does not bring significant improvement compared with the baseline mechanism as the chance of losing one packet is low and hence the benefit of Raptor coding is not evident. However, when $k = 204$, CeilingCast significantly improves the latency and throughput, by 20% to 70%, against the baseline mechanism, and the improvement grows with the number of luminaires. This significant improvement stems from the increased packet loss rate under a larger k, for which the benefit of Raptor coding becomes more evident.

TABLE I
PERFORMANCE UNDER MOBILE SCENARIOS

	CeilingCast		Non-Encoding	
k	Latency	Throughput	Latency	Throughput
26 (A&B)	0.51s	0.82kbps	0.50s	0.84kbps
204 (A)	4.19s	0.78kbps	7.21s	0.46kbps
204 (B)	5.27s	0.62kbps	11.27s	0.29kbps

We also perform evaluations under mobile scenarios where we deliberately let the phone cameras of two users (A and B) mis-capture one or two luminaires from time to time, and the results are reported in Table I. We have almost the same observations as those under stationary scenarios: whereas CeilingCast performs similarly to the baseline mechanism for $k = 26$, it significantly outperforms its competitor when $k = 204$: user A obtains an average throughput 0.78kbps with CeilingCast but only 0.46kbps otherwise, while user B obtains 0.62kbps against 0.29kbps otherwise. Therefore, these results again confirm the benefit of employing Raptor codes for enhancing reliability.

E. Power Consumption

The power consumption of CeilingCast has two parts: the consumption of LED control board and that of camera for

TABLE II
POWER OF CEILINGCAST.

	Transmitter (mW)	Receiver (mW)	Total (mW)
CeilingCast	~ 0	<287 [27]	<287
Wi-Fi	>800 [26]	50 [28]	>850

frame capture and decoding. Since CeilingCast utilizes an existing lighting infrastructure as transmitters, the consumption of the communication front-end (the luminaires) is actually zero. The real consumption is caused by receiving data from Ethernet or PLC backbone, as well as the encoding computations. Our field experiments show that the driver of CeilingCast has a power consumption of 40mW, but this consumption appears to be constant regardless of encoding computations, suggesting that it is mainly the consumption of driving light emission (the default function of a luminaire). Therefore, we can conclude that the tx power consumption of CeilingCast is negligible. As for Wi-Fi, existing Wi-Fi routers consume at least 800mW according to [25], [26].

The receiver of CeilingCast does lead to a rather high power consumption due to the use of image sensors. According to the consumption model for sequential frame capture in [27], we obtain the power $P \leq$ 177mW and $P \leq$ 287mW with $N = 1024 \times 768$ used by CV-based RoI extraction and $N = 3264 \times 2448$ used by lightweight RoI extraction, respectively. By using aggressive standby mechanism which lets the image sensor work in standby mode when no operation is performed [27], P drops to $P \leq$ 28mW and $P \leq$ 275mW respectively. Moreover if optimal clock scaling is adopted, the power can be reduced further. Wi-Fi appears to be more efficient at the receiver side by consuming slightly more than 50mW for data reception [28]. We summarize these quantities in Table II. In fact, as a data service, the transmitter side may constantly consume energy while the receiver side only causes consumption intermittently. Therefore, CeilingCast is much more energy efficient than Wi-Fi in reality.

VI. CONCLUSION

In this paper, we have presented CeilingCast as an LED-camera VLC system. It innovates in both encoding and decoding schemes to improve link reliability and throughput, so that it allows us to have the first realistic LED-camera VLC deployment, and it also provides us with practical insights on how such systems should be configured to reach its maximum capacity. Extensive field experiments have shown that our system can achieve a throughput much higher than a recent experimental prototype [19]. Our future work aims to further improve CeilingCast in terms of throughput and energy efficiency, by designing more effective coding/decoding schemes.

REFERENCES

[1] R. Drost and B. Sadler, "Constellation Design for Channel Precompensation in Multi-Wavelength Visible Light Communications," *IEEE Trans. on Antennas and Propagation*, vol. 62, no. 6, pp. 1995–2005, 2014.

[2] H. Burchardt, N. Serafimovski, D. Tsonev, S. Videv, and H. Hass, "VLC: Beyond Point-to-Point Communication," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 98–105, 2014.

[3] A. Ganick and D. Ryan, "ByteLight Scalable Indoor Location." [Online]. Available: http://www.bytelight.com

[4] N. Rajagopal, P. Lazik, and A. Rowe, "Visual Light Landmarks for Mobile Devices," in *Proc. of 13th ACM IPSN*, 2014, pp. 249–260.

[5] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta, "Luxapose: Indoor Positioning with Mobile Phones and Visible Light," in *Proc. of 20th ACM MobiCom*, 2014, pp. 447–458.

[6] C. Danakis, M. Afgani, G. Povey, I. Underwood, and H. Haas, "Using a CMOS Camera Sensor for Visible Light Communication," in *Proc. of 31th IEEE GLOBECOM*, 2012, pp. 1244–1248.

[7] N. Rajagopal, P. Lazik, and A. Rowe, "Hybrid Visible Light Communication for Cameras and Low-Power Embedded Devices," in *Proc. of 1st ACM VLCS*, 2014, pp. 33–38.

[8] L. Li, P. Hu, C. Peng, G. Shen, and F. Zhao, "Epsilon: A Visible Light Based Positioning System," in *Proc. of 11th ACM NSDI*, 2014, pp. 331–343.

[9] S. Rajagopal, R. Roberts, and S.-K. Lim, "IEEE 802.15.7 Visible Light Communication: Modulation Schemes and Dimming Support," *IEEE Communications Magazine*, vol. 50, no. 3, pp. 72–82, 2012.

[10] B. Zhang, K. Ren, G. Xing, X. Fu, and C. Wang, "SBVLC: Secure Barcode-Based Visible Light Communication for Smartphones," in *Proc. of 33rd IEEE INFOCOM*, 2014, pp. 2661–2669.

[11] T. Hao, R. Zhou, and G. Xing, "COBRA: Color Barcode Streaming for Smartphone Systems," in *Proc. of 10th ACM MobiSys*, 2012, pp. 85–98.

[12] S. Perli, N. Ahmed, and D. Katabi, "PixNet: Interference-Free Wireless Links using LCD-Camera Pairs," in *Proc. of 16th ACM MobiCom*, 2010, pp. 137–148.

[13] W. Du, J. C. Liando, and M. Li, "SoftLight: Adaptive Visible Light Communication over Screen-Camera Links," in *Proc. of IEEE 36th INFOCOM*, 2016.

[14] A. Wang, S. Ma, C. Hu, J. Huai, C. Peng, and G. Shen, "Enhancing Reliability to Boost the Throughput over Screen-Camera Links," in *Proc. of 20th ACM MobiCom*, 2014, pp. 41–52.

[15] W. Hu, J. Mao, Z. Huang, Y. Xue, J. She, K. Bian, and G. Shen, "Strata: Layered Coding for Scalable Visual Communication," in *Proc. of ACM 20th MobiCom*, 2014, pp. 79–90.

[16] W. Hu, H. Gu, and Q. Pu, "LightSync: Unsynchronized Visual Communication Over Screen-Camera Links," in *Proc. of 19th ACM MobiCom*, 2013, pp. 15–26.

[17] W. Huang and W. Mow, "PiCode: 2D Barcode with Embedded Picture and ViCode," in *Proc. of 19th ACM MobiCom*, 2013, pp. 139–242.

[18] T. Li, C. An, X. Xiao, A. T. Campbell, and X. Zhou, "Real-Time Screen-Camera Communication Behind Any Scene," in *Proc. of 13th ACM MobiSys*, 2015, pp. 197–211.

[19] H. Lee, H. Lin, Y. L. Wei, H. I. Wu, H. M. Tsai, and K. Lin, "RollingLight: Enabling Line-of-Sight Light-to-Camera Communications," in *Proc. of 13th ACM MobiSys*, 2015, pp. 167–180.

[20] A. Shokrollahi, "Raptor Codes," *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[21] H. R. Zeidanloo, "The Open Source Implementation of Raptor Code RFC5053." [Online]. Available: https://code.google.com/p/raptor-code-rfc/

[22] H. Ma, L. Lampe, and S. Hranilovic, "Integration of Indoor Visible Light and Power Line Communication Systems," in *Proc. of 17th IEEE ISPLC*, 2013, pp. 291–296.

[23] G. Corbellini, K. Aksit, S. Schmid, S. Mangold, and T. Gross, "Connecting Networks of Toys and Smartphones with Visible Light Communication," *IEEE Communications Magazine*, vol. 52, no. 7, pp. 72–78, 2014.

[24] "LUXEON XF-3535L." [Online]. Available: http://www.lumileds.com/products/matrix-platform/luxeon-xf-3535l

[25] "WiFi Router Power Consumption Database." [Online]. Available: http://www.tpcdb.com/list.php?type=11

[26] G. Palem and S. Tozlu, "On Energy Consumption of Wi-Fi Access Points," in *Proc. of IEEE 7th CCNC*, 2012, pp. 434–438.

[27] R. LiKamWa, B. Priyantha, M. Philipose, L. Zhong, and P. Bahl, "Energy Characterization and Optimization of Image Sensing toward Continuous Mobile Vision," in *Proc. of 11st ACM MobiSys*, 2013, pp. 62–82.

[28] A. Carroll and G. Heiser, "An Analysis of Power Consumption in a Smartphone." in *Proc. of USENIXATC ATC*, 2010.