POLI: Long-Range Visible Light Communications Using Polarized Light Intensity Modulation

Chun-Ling Chan[†], Hsin-Mu Tsai[†], Kate Ching-Ju Lin[‡]
[†]Department of Computer Science and Information Engineering, National Taiwan University, Taiwan
[‡]Department of Computer Science, National Chiao Tung University, Taiwan
r03922036@ntu.edu.tw, hsinmu@csie.ntu.edu.tw, katelin@cs.nctu.edu.tw

ABSTRACT

Recent studies have demonstrated the potential of light-to-camera communications for realizing applications such as localization, augmented reality and vehicle-to-vehicle communications. However, the fundamental requirement of visible light communications, *flicker-free illumination*, becomes a key limitation hindering existing technologies from serving a camera at larger distances. To break this limitation, this paper presents POLI, a light-to-camera communication system that exploits a novel *POlarized Light Intensity modulation* scheme to provide reliable communications for a wide range of distances. The key idea of POLI is to hide the information with the polarization direction of the light, to which human eyes are insensitive. Taking advantage of this, POLI can change the intensity of the polarized light as slowly as possible, at a rate determined by the range the system would support, but does not generate flickers. By using an optical component to "transform polarization directions to colors", POLI allows a camera to leverage its received RGB values as the three spatial dimensions to recover the information carried in different data streams. POLI further incorporates a number of designs to tackle the non-linearity effect of a camera, which is especially critical for an intensity-based modulation scheme. We implemented a prototype using the USRP N200 combined with off-the-shelf optical components. The experimental results show that POLI delivers to its camera receiver a throughput proportional to the dynamic channel conditions. The achievable throughput can be up to 71 bytes per second at short distances, while the service range can be up to 40 meters.

1. INTRODUCTION

Light-to-camera communications has recently become an emerging network paradigm. The potential applications that greatly benefit from such a system include advanced driving assistance system (ADAS) and augmented reality. For example, real-time status information about a particular vehicle, such as its speed, bearing, and brake status, can be emitted by its LED tail lights, or users can capture the information required for augmented reality based navigation in a large museum from ceiling lights. The key to the success of the above applications is: communication range! However, most existing designs [9, 23, 17, 18] only support a room-level communication range, limiting their applicability.

The main challenge of enabling long-range light-to-camera communications is to combat with severe path losses. Essentially, the intensity of a pixel in an image is the amount of optical energy accumulated by the sensing area of that pixel within the exposure duration, i.e., when the shutter is open. Hence, a camera at longer distances experiences a larger path loss and no longer accumulates sufficient energy for reliable decoding. To extend the service range, a simple solution is to extend the exposure time and, thereby, increase the signal strength. Unfortunately, there exists a limit of the longest exposure time a camera can use. In particular, any exposure time longer than a symbol duration could cause *inter-symbol interference*, meaning that a camera would receive mixtures of consecutive symbols in an image, making all of the mixed symbols undecodable. One might think that, to avoid symbol mixtures, we may further enlarge the symbol duration accordingly, which, however, could lead to observable flickers. This, hence, sets a hard limit for a non-flickering light to enable long-range and reliable communications.

In this paper, we introduce POLI, a light-to-camera communication system that uses *POlarized Light Intensity modulation* to overcome the above fundamental limitation, allowing a distant camera of a few tens of meters to still receive reliably. The key idea of POLI is to circumvent the limitation set by the flicker-free requirement using a novel optical design that leverages an extra dimension – polarization. To do so, instead of directly modulating a light source, we modulate the polarized lights along different directions, while still maintaining the emitted light intensity stable all the time. Since human eyes are insensitive to polarization, there no longer exist any limitation on the symbol duration and, thereby, the exposure time a camera can use. To realize such a system, however, a few key questions need to be answered.

First, how do we hide observable flickers, even when the signal has significant energy at lower frequency? POLI tackles this challenge by a novel polarization intensity modulation scheme. The scheme creates a signal such that the total intensity of light in different polarization directions is kept constant over time, while the intensity of each polarized light can be changed independently even at a very low frequency. On the other hand, polarization of light also serves as a tool to multiplex different data streams and boost the data rate.

Then, how exactly should a camera recover the information embedded in different polarizations of light? A camera typically accumulates the intensity of all the polarizations of light, but now needs to detect intensity changes along some particular polarizations of light. To do so, inspired by [35], POLI makes use of an optical rotatory dispersor to *transform polarization direction into colors*. Since

MobiSys'17, June 19-23, 2017, Niagara Falls, NY, USA

ISBN 978-1-4503-4928-4/17/06...$15.00

DOI: http://dx.doi.org/10.1145/3081333.3081352

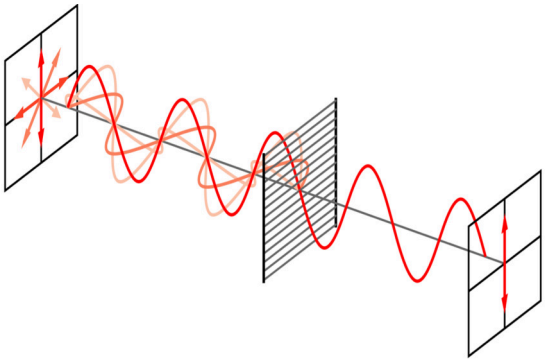


Figure 1: **Optical polarization.** Unpolarized light can be converted to polarized light after passing the first polarizer, which is then partially blocked by the second polarizer.

a commodity camera captures the red, green, and blue (RGB) values at each pixel of an image, these three color components can be thought of the spatial dimensions of a camera, which behaves like a three-antenna device that has three degrees of freedom and can separate three simultaneous data streams. Therefore, by attaching a polarizer, a camera receiver should be able to observe colors and recover information modulated in different polarized color beams.

Finally, how does the system deal with non-linearity of a camera? This non-linearity effect would introduce two practical problems. First, most existing MIMO decoders are linear equalizers, which would produce large decoding errors if the channels are non-linear. Second, each camera has a limited dynamic range: incoming light waves saturate the output at short distances and generate low pixel values with little changes at long distances. We solve the first problem by a new channel estimation scheme that improves decoding reliability even for non-linear channels. To further alleviate the impact of non-linearity, we propose an automatic gain control scheme that adapts the exposure time to dynamic channels such that the received intensity can be mostly within the linearity range of a camera, thereby enabling reliable communications in a wide range of distance.

We have built a prototype of POLI using the USRP N200 radio platforms to control the three LEDs and generated intensity varying polarized lights, which are then converted to color beams by a dispersor. A Point Grey Grasshopper camera is used as the receiver to measure the RGB values of the combined color beam and perform MIMO decoding. The performance of POLI is evaluated via testbed experiments, and our key findings are as follows:

- The intensity of the combined white light of the transmitted polarized color beams is nearly constant, thereby indistinguishable to human eyes. However, by installing a polarizer, a camera can clearly detect the variation in its received RGB values and use MIMO decoding to separate the concurrent streams modulated in different color beams.
- Each exposure time has its own optimal operational range. Without proper configuration, the linearity of the received RGB intensity might be too poor to ensure reliable decoding. By enabling POLI's gain control, a receiver can quickly identify its suitable exposure time and achieve a symbol delivery ratio very close to the one produced by the optimal configuration.
- POLI provides a camera a throughput proportional to its channel condition. As the baud rate is set to 30 symbols per second, the effective throughput at distances 5 m and 10 m are, on average, 34 and 9 bytes per second, respectively. The service range can even be extended to 40 meters with an error rate of only 10%.
- The service range is determined by the baud rate (i.e., inverse of the symbol interval) used by a transmitter. With POLI, a transmitter can flexibly reduce its baud rate in order to extend its service range. The smallest baud rate is now no longer limited by a lowest frequency required by existing light-to-camera communication systems to prevent observable flickering, and can be reduced to 5 Hz or less.

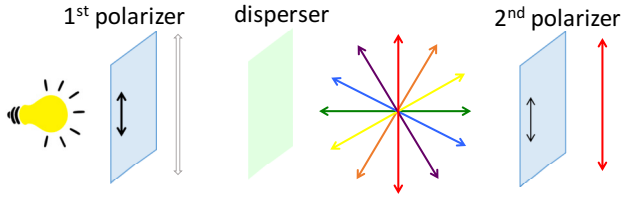


Figure 2: **Transmitting a color beam via dispersion.** The dispersor diverts polarized light from the first polarizer into several color beams in different polarizations; then, the second polarizer only allows a polarized color beam to traverse.

2. PRIMER

2.1 Polarization

Most common sources of visible light, such as LED, typically emit unpolarized light, which consists of a mixture of waves with different polarizations. To generate polarized light, one can pass unpolarized light through an optical polarizing filter or simply "polarizer", as shown in Fig. 1. A polarizer[1]. This polarization effect will not be perceived by users since human eyes cannot detect polarization directions. To measure the oscillating direction of polarized light, we can pass the light source through another polarizing filter, as shown in Fig. 1. Then, the intensity of the beam traversing through the second filter will be determined by the angle θ between the directions of the two polarizers. According to the Malus's law [8], the intensity of the beam passing through two polarizers can be given by

$$I = I_0 \cos^2 \theta, \tag{1}$$

where I_0 indicates the original light intensity. From the equation, we get that, ideally, the light can completely pass through the filters if the directions of the two polarizers are in parallel. However, the light is fully blocked if one filter is perpendicular to the other.

2.2 Dispersion

Dispersor is an optical material that can separate a white light into components of different wavelengths (i.e., different colors) and rotate them to different polarization directions. If we simply place a dispersor in front of a light source, human eye or image sensors still see the white light since all those dispersed polarized color beams are combined together. A possible way to emit a color beam is to place a dispersor in between two linear polarizers, as shown in Fig. 2. The first polarizer makes unpolarized light become linearly

[1]Polarizers can further be classified into *absorptive polarizers* and *beam splitting polarizers* allows only a particular polarization to pass through, while blocking the waves oscillating in all the other directions. The former absorbs the unwanted polarizations, while the later reflects the unwanted polarizations. We use absorptive polarizers in our work and simply refer to the term "polarizer" as the absorptive polarizer.

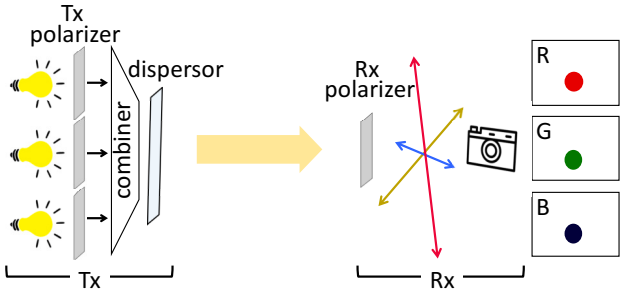


Figure 3: **POLI's design.** Light from the three LEDs is diverted into color beams along different polarization directions. By filtering them through another polarizer, the receiver only senses three color beams aligned in different polarizations, which can be recovered using MIMO decoding technologies.

polarized light. The dispersor then diverts the polarized light into several color beams (each with a distinct wavelength) in different polarization directions [35]. Since human eyes cannot detect polarizations, without the second polarizer, people still see the white light, though the light intensity might be slightly lowered after dispersion. However, by attaching the second polarizer to a receiving image sensor (or camera), the receiver only receives one particular color beam that aligns along the direction of its polarizer and can, hence, traverse through. By combining a dispersor with two polarizers, we can tune the direction of either polarizer to change the color of the light beam and explicitly control the RGB intensity of the image pixels occupied by a transmitting light. This gives us an opportunity to realize intensity-based modulation, which is more suitable for long-range VLC, without causing flickers, and the information can easily be recovered by measuring the received RGB intensity. We will explain in Sec. 3 why such polarization intensity modulation provides reliability even at a long distance.

2.3 Intensity Modulation

Intensity modulation represents digital data as variations in the intensity of a light source. The basic operation is intrinsically the same with traditional *amplitude modulation* used in RF-based systems. The only difference is that a light source can only emit non-negative amplitude and, hence, should represent data using light intensity, i.e., the amount of energy. For example, for M-ary intensity modulation (M-IM), each symbol is assigned one of M different intensity values, which are usually uniformly distributed between the maximum intensity $I_{\max}$ and minimum intensity $I_{\min}$ as follow:

$$I_m = \frac{I_{\max} - I_{\min}}{M-1}(m-1) + I_{\min}, \forall m = 1, \cdots, M. \quad (2)$$

Each intensity is then used to represent a unique sequence of $\log_2 M$ digital bits. Hence, the ideal data rate of intensity modulation equals to $\frac{\log_2 M}{T}$, where T is the symbol duration. Such intensity modulation is more resilient to small occupied areas of a light in an image or can even work for single-pixel light sensors. However, varying intensity is intrusive as human eyes will perceive flickers. Therefore, our goal is to develop a system that can alternatively vary the intensity of each color component, while stabilizing the aggregated intensity of color beams.

3. DESIGN

3.1 Overview

The key of POLI is to develop a novel modulation technique, called *polarization intensity modulation* (PIM). To avoid flickering while maintaining the nice properties of intensity modulation, POLI's PIM represents digital bits using the intensity of RGB values received by a camera or light sensor, instead of the intensity of the transmitting light. To do so, we combine three LED light bulbs and a dispersor as a transmitter, as shown in Fig. 3. Each LED is attached a linear polarizer that converts unpolarized light to polarized light and operates like an *antenna* in a traditional MIMO wireless device. To ensure that each light source can be converted to a different color beam, we equip the three polarizers along three different directions. The three light beams are then aggregated via a beam combiner, which passes the combined light through the dispersor for diverting the color beams along different directions.

POLI's receiver can be either a rolling shutter camera, a global shutter camera or a color sensor. To detect color intensity modulated by the transmitted light, the receiver attaches another polarizer in front of its lens. If only a single LED is transmitting, the camera will only receive a single color beam, as shown in Fig. 2.[2] However, since POLI combines three LEDs, each with a polarizer along a different direction, the camera thus receives the combination of the three different color beams, each of which is from a transmitting LED light, as shown in Fig. 3. By changing the intensity of each LED, and thereby the intensity of each color beam, the camera will receive varying RGB values in those pixels occupied by the combined color beam (see Sec. 3.2). Note that a camera now receives in three dimensions, i.e., three RGB intensity values. Hence, POLI's design is analogous to a 3×3 MIMO system and can further leverage the multiplexing gain to boost the data rate. Specifically, a 3-antenna MIMO system can typically deliver three data streams simultaneously. In POLI, the intensity variation of the three LEDs can be thought of three data streams, which can be recovered from the three received RGB values (see Sec. 3.3) using MIMO decoding algorithms.

However, since the color intensity received by a camera not only depends on the transmitted light intensity, but also its configured exposure time. If the exposure time is not properly selected, the received RGB intensity might saturate due to over exposure or become too low due to under exposure, as a result leading to non-linear channel response. This non-linearity effect makes standard linear MIMO equalizers ineffective. To resolve this non-linearity problem, we propose a table lookup decoding method in Sec. 3.3 to recover the combined signals, and further develop an automatic gain control scheme in Sec. 3.4 to adapt the exposure time of a receiver to dynamic channel conditions so as to enlarge the linearity range.

3.2 Polarization Intensity Modulation

The proposed *Polarization Intensity Modulation* (PIM) is a technique that varies the intensity of a polarized light beam, instead of the aggregated light intensity. In particular, each LED changes its light intensity over time, while the intensities of the three lights are well-controlled such that the total intensity output by the beam combiner can still be a constant. Consider a simple case where only a single LED light bulb i is transmitting using a unit of intensity. The dispersor diverts the light into multiple color beams, but only one of the beams passing through the receiver's polarizer is captured in the image. Let $\mathbf{c}_i = (R_i, G_i, B_i)$ denote the RGB inten-

[2]For ease of description, we mainly consider the scenario with a single transmitting light. However, the design can be easily extended to a scenario with multiple transmitters if the receiver is a camera, which can leverage its high spatial resolution of image sensors to separate multiple transmitters at different locations.

sity of the received color beam captured in the image when LED i transmits signals with a unit of intensity. This color intensity vector $\mathbf{c}_i$ can be thought of a unique channel between the i-th transmitting light and the receiving camera, similar to the channels of a 1×3 MIMO link. To simplify the description, for now we temporarily assume that the received intensity of each RGB component is linearly proportional to the intensity of the transmitted light. In other words, if the transmitting LED changes its light intensity to x, the received color vector becomes $(R_i, G_i, B_i)x$. In POLI, each LED i, $i = 1, 2, 3$, sets its intensity to $x_i(t)$ at time t. The camera then receives the combination of the three color beams, and the RGB values of each pixel occupied by the light beam can be expressed by

$$\begin{pmatrix} y_R \\ y_G \\ y_B \end{pmatrix} = \begin{pmatrix} R_1 \\ G_1 \\ B_1 \end{pmatrix} x_1 + \begin{pmatrix} R_2 \\ G_2 \\ B_2 \end{pmatrix} x_2 + \begin{pmatrix} R_3 \\ G_3 \\ B_3 \end{pmatrix} x_3. \tag{3}$$

To ensure that the above three equations are linearly independent, the polarizers in front of three lights should be attached along three different directions. We place them in 10°, 60° and 100° in our implementation, but any configuration of three independent directions works. Note that the intensity of the combined light beam should keep constant. Hence, the following total intensity constraint should always hold:

$$x_1 + x_2 + x_3 = I_{\text{total}}, \tag{4}$$

where I_{total} is the constant light intensity of POLI's transmitter. Subject to this constraint, we use x_1 and x_2 as data symbols, but use x_3 as a *complement* to satisfy Eq. 4. That is, similar to power control in most of RF-based systems, POLI should also perform *intensity control* by setting x_3 to $I_{\text{total}} - x_1 - x_2$. Hence, Eq. 3 can be rewritten as the following vector form:

$$\mathbf{y} = \mathbf{c}_1 x_1 + \mathbf{c}_2 x_2 + \mathbf{c}_3 (I_{\text{total}} - x_1 - x_2), \tag{5}$$

where $\mathbf{c}_i$ is the unique channel vector between the i-th transmitting LED and a receiving camera.

Due to this intensity constraint, though POLI consists of three LED light bulbs, there are, however, only two lights available for carrying data signals x_1 and x_2. In particular, we call the first two LEDs the *data light sources* and call the last LED the *complementary light source*. Then, for each of the data light sources, we use an M-ary PIM (M-PIM) symbol to represent any $\log_2 M$ bits of information. The intensity of each M-PIM symbol I_m for $m = 1, \cdots, M$ is given by

$$I_m = \frac{I_{\text{total}}/2 - I_{\min}}{M - 1} * (m - 1) + I_{\min}, \text{for } m = 1, \cdots, M, \tag{6}$$

where $I_{\text{total}}/2$ is the maximum intensity allocated to a data light source and $I_{\min}$ is the minimum intensity. Then, the combination of the intensity of two light bulbs, i.e., $\mathbf{x} = (x_1, x_2)$ at time t represents a unique pattern of $2\log_2 M$ bits, and we collect all the combinations as the symbol set $\mathbb{S}$. For example, if POLI adopts 16-PIM, each light bulb encodes four bits into 16 intensity levels. Then, the symbol set $\mathbb{S}$ includes 256 symbol vectors, and any transmitted symbol vector can be expressed by

$$\mathbf{x}(t) = (x_1(t), x_2(t)) = \begin{cases} (I_1, I_1) & \text{for bit pattern 1} \\ (I_1, I_2) & \text{for bit pattern 2} \\ \vdots & \\ (I_1, I_{16}) & \text{for bit pattern 16} \\ (I_2, I_1) & \text{for bit pattern 17} \\ \vdots & \\ (I_{16}, I_{16}) & \text{for bit pattern 256.} \end{cases} \tag{7}$$

Then, the camera receives one of the 256 possible RGB intensity vectors and maps it back to the digital bits.

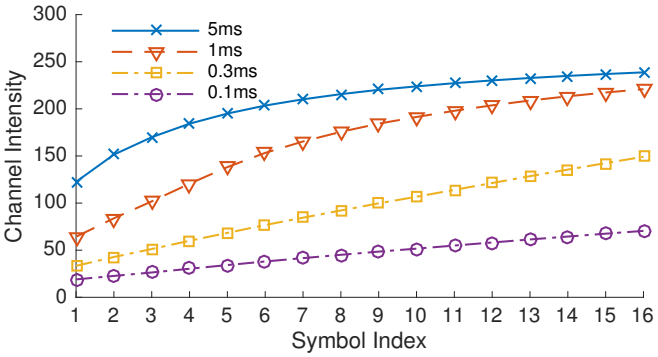


Figure 4: **Non-linearity of intensity-modulated signals.** A camera usually has a poor linearity, especially when the exposure time is not set properly. This non-linearity effect hinders us from using the conventional linear MIMO equalizer, e.g., zero-forcing, to recover the signals.

3.3 Channel Estimation and Demodulation

Due to the analogy between our POLI and a 3×3 MIMO system, in theory, we can recover the three unknowns, x_1, x_2 and x_3, in Eq. 3 using the standard linear MIMO decoders, such as the *zero forcing equalizer* [29]. However, the feasibility of linear equalizers relies on an assumption of *signal linearity*, which refers to as the property that the received signal should be linearly proportional to the transmitted signal. However, in a visible light communication system, the image intensity is determined by not only the intensity of the transmitting light but also camera sensitivity or configurations, such as exposure time. For example, if the exposure time is set to too long, the RGB intensity is very likely to saturate, i.e., close to 255. Similarly, if the exposure time is too short, the received RGB intensity would be very close to 0 most of the time. Hence, a VLC system usually is not a perfect linear system. To verify this point, we measure the RGB intensity received by a camera deployed at distance 1 m when only the first LED modulates its signals using 16-Ary PIM (i.e., 16 different intensity values). The maximum and minimum transmitted intensity, $I_{\max}$ and $I_{\min}$, are set to 35 and 150, respectively. Fig. 4 only plots the intensity of the blue color when the exposure time is set to four different values. The results show that, for some configurations of the exposure time, e.g., 5 ms, the received intensity quickly saturates when the transmitter increases its amplitude. That is, an improper exposure time could lead to poor linearity. If this is the case, decoding the received symbols using traditional linear equalizers could produce a high error rate.

Table 1: **Intensity-to-symbol lookup table.**

Rx RGB Intensity $f(\mathbf{x})$	Tx symbol $\mathbf{x} = (x_1, x_2)$
$(R_{1,1}, G_{1,1}, B_{1,1})$	(I_1, I_1)
$\vdots$	$\vdots$
$(R_{1,M}, G_{1,M}, B_{1,M})$	(I_1, I_M)
$(R_{2,1}, G_{2,1}, B_{2,1})$	(I_2, I_1)
$\vdots$	$\vdots$
$(R_{2,M}, G_{2,M}, B_{2,M})$	(I_2, I_M)
$\vdots$	$\vdots$
$(R_{M,M}, G_{M,M}, B_{M,M})$	(I_M, I_M)

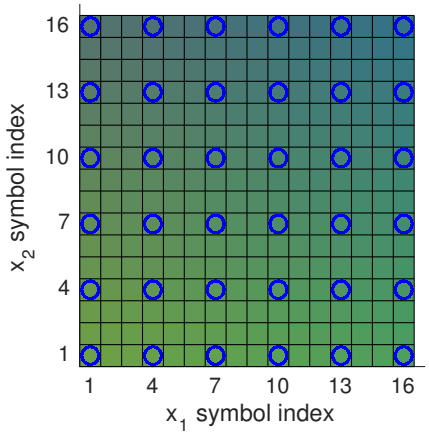


Figure 5: **Channel estimation via piecewise bilinear interpolation.** We select the symbol vectors (the blue circles) at the corners of a grid as training symbol vectors, and estimate the received RGB intensity of the other symbol vectors (the color of each block) within a grid using bilinear interpolation.

To resolve this non-linearity issue, we exploit a table lookup decoding scheme to recover the multiplexing streams. In particular, instead of learning the channels (i.e., the intensity vectors $\mathbf{c}_i$ from LED i) and decoding via linear MIMO equalizers, we alternatively allow each receiver to measure the intensity-to-symbol table that directly maps the received RGB intensity values to the most likely transmitted symbol. Such table lookup decoding is simple but effective mainly because, in POLI, though the received RGB intensity is non-linear, it, however, does not change is quite stable since the optical fading is much more static as compared to the fading of RF signals. Hence, the lookup table does not need to be updated frequently if the light-to-camera link distance does not change much.

To realize table lookup decoding, POLI's transmitter sends some known training symbols in the beginning of a packet for each receiving camera to update its intensity-to-symbol lookup table. To reduce the training overhead, the transmitter only samples a small subset of symbol vectors (I_i, I_j) from $\mathbb{S}$ as the training symbol vectors. A camera measures the received RGB intensity of the training symbol vectors, as the blue entries shown in Table 1, and leverage *piecewise bilinear interpolation* to estimate the RGB intensity of the remaining symbol vectors, as the black entries shown in Table 1. By *piecewise bilinear interpolation*, we mean that the estimated RGB intensity of a non-training symbol vector is learned from taking bilinear interpolation of a few closest training symbol vectors. By collecting the intensity of a sufficient number of training symbol vectors, such piecewise linear interpolation should approximately fit the actual non-linear channels, e.g., the channels shown in Fig. 4. In particular, we adopt grid sampling to select the training symbol vectors. In particular, the set of the selected training symbol vectors can be expressed as

$$\mathbb{S}_{\text{train}} = \{(I_i, I_j) : 1 \le i, j \le M, i \bmod d{=}1, j \bmod d{=}1\}, \quad (8)$$

where d denotes the sampling interval, meaning that each data LED samples one out of every d possible intensity values as a training known symbol. Then, we can adjust the interval d to determine the density of training symbols. Given the measured RGB intensity of the training symbol vectors, the receiver performs *bilinear interpolation* for the non-training symbol vectors within a rectilinear 2D grid formed the four closet training symbol vectors, as illustrated in Fig. 5. For example, for the symbol vectors (I_i, I_j), $1 \le i \le d, 1 \le j \le d$, their RGB intensity intensity is estimated by bilinear interpolation of the measured RGB intensity of the four training symbol vectors, $(I_1, I_1), (I_1, I_d), (I_d, I_1)$ and (I_d, I_d). Since each grid performs its interpolation independently, this piecewise interpolation can alleviate the non-linearity problem when the training symbol vectors are sampled dense enough. We will show in §5.1 that 16 training symbols are sufficient to produce a fairly good decoding reliability. As compared to traditional linear equalizers, which also need to send a few training symbols from each antenna, POLI only needs slightly higher channel estimation overhead, but can effectively resolve the non-linearity problem.

Given the intensity-to-symbol lookup table, a receiver can recover its symbols using ML (Maximum Likelihood) decoding. In particular, let $\mathbf{x} = (x_1, x_2) \in \mathbb{S}$ denote any possible symbol vector sent by the transmitter's two data LEDs and let $f(\mathbf{x})$ be the measured or interpolated RGB intensity corresponding to $\mathbf{x}$. Then, ML detection of a symbol vector can be written as

$$\hat{\mathbf{x}} \quad = \quad \arg\min_{\mathbf{x} \in \mathbb{S}} \|\mathbf{y} - f(\mathbf{x})\|^2, \quad (9)$$

where $\mathbf{y}$ is the received RGB intensity vector. Though enabling ML requires a receiver to search the whole space, this complexity is, however, acceptable for a VLC system, in which the bandwidth (i.e., the frame rate) of a receiving camera is usually much smaller than RF devices, giving a camera receiver a long enough time to decode every symbol.

Detailed protocols of POLI. To realize the above ML detection in practice, we collect several data symbols as a packet, and broadcast a delimiter and the training symbols in the beginning of each packet, as shown in Fig. 6. The delimiter consists of 20 known symbols that are used for a receiver to detect the beginning of a packet. The delimiter is followed by the training symbols for a receiver to update its intensity-to-symbol table. Since different receiving cameras usually have different frame rates, a receiver's sampling interval is usually different from the symbol duration of the transmitted light signals. Therefore, the transmitter should configure its symbol duration such that any receiver can at least receive each symbol in at least one image frame. To improve reliability, we let the symbol baud rate be four times of the frame rate of a receiving camera in our implementation. With this configuration, a receiver can receive a symbol across four image frames and take the average intensity to perform table lookup decoding. One might doubt that a receiver would receive the mixture of two consecutive symbols in an image frame if a receiver is not tightly synchronized with its transmitter. This is however rarely happens since a camera does not receive the light signals all the time, but only during the exposure time, which

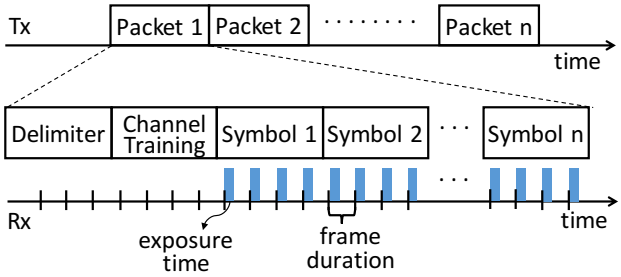


Figure 6: **Protocol and packet format.** Each packet preceded with a delimiter and a few channel training symbols for a receiver to detect the packet and train its lookup table.

is actually much shorter than a frame interval, as shown in Fig. 6. Therefore, the probability that a symbol boundary happens to locate within an exposure time is fairly small.[3]

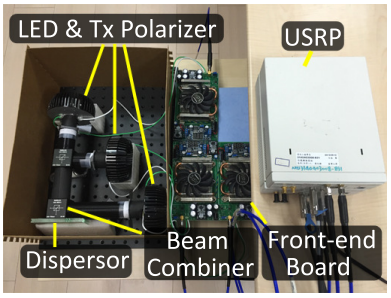
3.4 Automatic Gain Control

Though POLI leverages table lookup decoding to alleviate the impact of non-linearity, the symbol error rate might still be high if the channel is extremely non-linear since the distance between the intensity RGB intensity $f(\mathbf{x})$ of any two neighboring symbol vectors might become too small. To improve the decoding success probability, a receiver should adapt its exposure time to dynamic environmental conditions, such as the distance to the transmitter and the strength of the ambient light. More specifically, to minimize the symbol error rate, a receiver should search the best exposure time such that the received intensity of POLI's PIM signals can be mostly within the linearity range of the receiver. This task is fundamentally the same with *automatic gain control* (AGC) required in most of RF-based communication systems.

To achieve this goal, we adopt a simple adaptation scheme to search the best configuration during data reception. The high-level idea is to decrease the exposure time if the maximum received RGB intensity is too large, while increasing it if the minimum intensity is too small. In particular, a receiver maintains a window of K frames and keeps tracking its maximum and minimum received intensity among the pixels occupied by the light in those K frames. For each pixel, we use $\max(R, G, B)$ to represent its intensity as searching for the maximum intensity, while using $\min(R, G, B)$ to represent its intensity as looking for the minimum intensity. The receiver switches to the next longer exposure time if the minimum intensity is lower than a threshold $P_{\min}$, while switching to the next shorter exposure time if the maximum intensity is higher than a threshold $P_{\max}$. We set $P_{\max} = 255 * (1 - \alpha)$ and $P_{\min} = 255 * \beta$, where α and β are small margins used to prevent saturation and low intensity, respectively. In our experiments, the two values are heuristically chosen without much efforts, but can usually ensure quick convergence for static scenarios.

The last thing worth noting is that the longest feasible exposure time is bounded by the symbol duration used by the transmitter. Otherwise, if the exposure time is longer than a symbol duration, the receiver will expose two or more consecutive symbols in an image frame, leading to inter-symbol interference. In other words, since a farther receiver usually receives a lower intensity and needs to use a longer exposure time, to extend POLI's service range, we should enlarge the symbol duration. This, however, reduces the achievable data rate. We will experimentally evaluate this tradeoff in Sec. 5.

[3]If, unfortunately, the exposure time really overlaps with the symbol boundary, we leverage the delimiter to detect the symbol boundary and remove those mixture frames.



(a) Transmitter

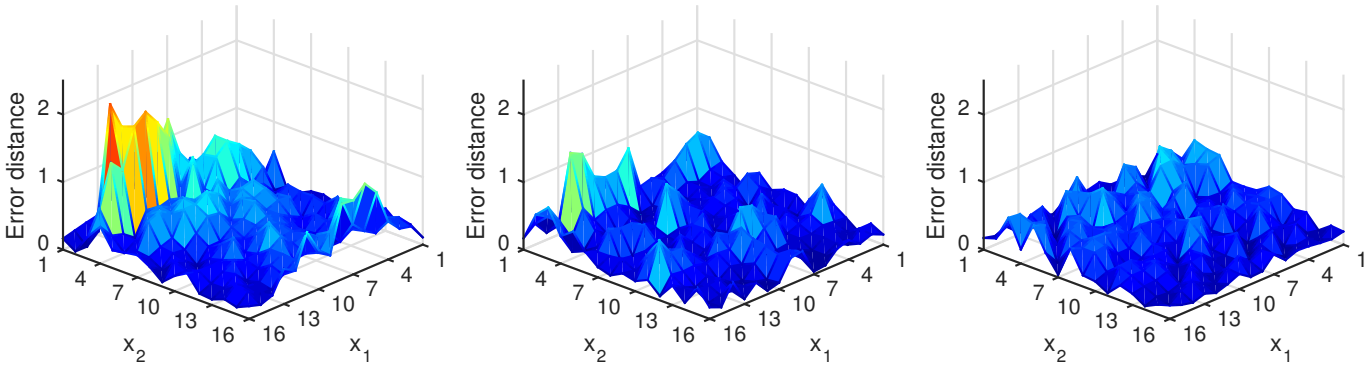
(b) Receiver

Figure 7: **POLI prototype.**

4. IMPLEMENTATION

We combine three Bridgelux BXRA-W0802 12.5 watt LED light bulbs as POLI's transmitter. Each LED light is controlled by an Ettus USRP (Universal Software Radio Peripheral) N200 board [1] through a VLC front-end board, as shown in Fig. 7(a). In particular, the USRP board is equipped with a LFTX daughterboard that can generates voltage varying signals with a sampling rate between 0 and 30 MHz. The voltage signals are then converted to the current varying signals by the front-end board to drives an LED light. The intensity of a transmitting LED can be tuned by adjusting the voltage of the signals sent by the USRP controller. Each LED is controlled by one USRP board, and the three USRPs are synchronized through an external clock [2] so that they can update their intensity at exactly the same time. A polarizer is attached to each of the three LEDs, and the directions of the three polarizers are aligned to 10, 60 and 100 degrees, respectively. The three light beams are then aggregated by a Thorlabs CM1-BS013 beam combiner [3], which passes the combined light beam through an EDC7723 LCD dispersor.

At the receiver side, we use a Point Grey Grasshopper camera with a global shutter that outputs the received optical signals as image frames with a resolution of 1920×1200, as shown in Fig. 7(b). The Grasshopper camera provides flexibility to tune the parameters, including the frame rate, the exposure time and the gain. Without otherwise stated, we let the camera receive a few random symbols and adapt its gain accordingly before data reception. The captured images are saved as the JPG or RAW formats for offline processing and decoding in a Linux system. Though the evaluation is done offline, we collect the traces over real channels to evaluate the performance with consideration of channel variation and the impact of imperfect channel estimation.



(a) Linear sampling (JPG image) (b) Grid sampling (JPG image) (c) Grid sampling (RAW image)

Figure 8: **Error distance per symbol.** The distribution of error distances is non-uniform because of the non-linearity effect inherently in a VLC system. The error could be large for some intensity regions, as shown in (a), if the symbols are decoded using a linear equalizer. Our ML-based decoding alleviates the impact of non-linearity and reduces the decoding error, as shown in (b). The errors can further be reduced if a camera outputs RAW images, which do not distort the received information, as shown in (c).

Parameter	Options	Default
Transmission baud rate	5,15,30,60 (Hz)	30 (Hz)
Camera frame rate	20,60,120,240 (fps)	120 (fps)
Exposure time	0.08,1,7,16,30 (ms)	7 (ms)
Modulation order	8,16,32	16
Min/Max intensity in PIM, $I_{\min}$, $I_{\max}$	-	35,150
Training symbol sampling distance d	3,5,15	3
Threshold for AGC, $P_{\min}$, $P_{\max}$	-	80,150
Image format	JPG, RAW	RAW

Table 2: **Parameters and default values.**

5. RESULTS

We conduct testbed experiments to characterize the benchmark performance of POLI, including the channel estimation error, the required channel training cost, the impact of channel correlation and exposure time configuration. For testing a more challenging case in indoor environments, we experiment when all interfering lights, three light fixtures, each with 4 14W T5 fluorescent tubes, are turned on during normal daylight. They would generate ambient light which adds noises to the received signal. We also conduct some experiments in an outdoor environment during the daytime to test how POLI works in this more challenging scenario. The performance of POLI is evaluated in terms of the *i) symbol delivery ratio*, which is defined as the number of correctly delivered symbols divided by the total number of transmitted symbols, and the *ii) effective throughput*, which is defined as the symbol rate multiplied by the symbol delivery ratio. Other default parameter settings are listed in Table 2.

5.1 Micro benchmark

5.1.1 Demodulation Errors

We first check the effectiveness of our table lookup decoding with grid-based training and compare it with traditional linear equalization.

Experiment setup: A receiving camera is deployed at a distance 1 m away from the transmitter, which modulates the signals using 16-Ary PIM. We compare our table lookup ML decoding with the traditional ZF decoding. For our ML decoding, the transmitter adopts grid sampling and set the sampling interval d (in Eq. 8) to 5. For ZF decoding, we randomly select 12 known training symbol vectors for a receiver to train its channel vectors $\mathbf{c}_i, i = 1, 2, 3$. The transmitter first broadcasts the training symbols and then transmits all the 256 possible symbols iteratively. For each experiment, the training symbols are only sent once, while we repeat symbol transmissions for four iterations. To check the impact of channel non-linearity on decoding errors, we calculate the error distance between the true received intensity and the estimated intensity (i.e., $f(\mathbf{x})$ in the table for ML and $\sum_i \mathbf{c}_i x_i$ for ZF). Since the intensity information might be distorted after image compression, we further check whether such compression would affect the error distance by saving the frames as the JPG and RAW formates, respectively.

Results: Fig. 8 illustrates the error distance, i.e., the distance between the expected color intensity estimated by the learned channels and the actual measured color intensity, for ZF (JPG), ML (JPG) and ML (RAW), respectively. The results show that the error distances of ZF are much larger than that of our table lookup decoding. For ZF, the errors for symbols with a larger index, i.e., those corresponding to a higher transmitted intensity, are much larger. This is because, as shown in Fig. 4, the non-linear effect usually occurs in the high intensity region when the receiving camera uses an exposure time that causes overexposure. By enabling our table lookup ML decoding with grid-based sampling, the impact of non-linearity can be minimized as the bilinear interpolation is performed only within a grid, instead of the whole symbol vector space. We can also observe that the RAW image format, as shown in Fig. 8(c), produces a much smaller error than the JPG image format, as shown in Fig. 8(b). This explains that image compression introduces additional noise and further worsens the non-linearity problem.

5.1.2 Channel Training Cost

Since the training symbols sent per-packet are an overhead of the system, we next check how many training symbols are sufficient for a receiver to achieve reliable decoding.

Experiment setup: We use the same sampling strategy as the previous experiment to select the training symbols for 16-ary PIM, but

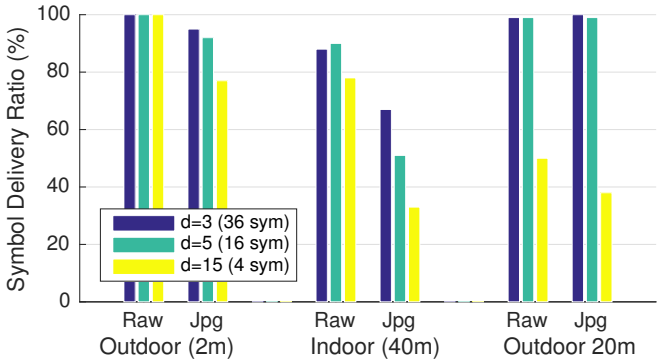


Figure 9: **The impact of the training cost on the symbol delivery ratio.** When the channels are less linear, e.g., as the link distance becomes longer, more training symbols, i.e., a smaller sampling interval, are required to perform piecewise bilinear interpolation and approximate the real channels. From our measurements, training symbols are dense enough to achieve accurate channel estimation as $d = 5$.

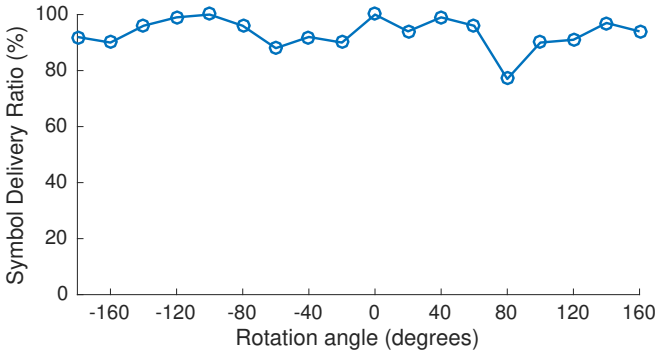


Figure 10: **Impact of rotation on channel correlation.** Similar to most of MIMO systems, the practical achievable multiplexing gain is determined by the correlation between the channel vectors. POLI achieves a high symbol delivery ratio, except for some worse spots with a polarization direction lead to a higher channel correlation.

now try different sampling densities. In particular, each LED selects one training symbol out of every $d = 3$, 5 and 15 symbols, respectively. Then, the RGB intensity of non-training symbols within a grid is trained via bilinear interpolation according to the measured intensity of the four corner symbol vectors. The transmitter sends 100 randomly-generated data symbols after broadcasting the training symbols, while the receiver decodes the data symbols and calculates the symbol delivery ratio, which is defined as the number of the correctly recovered symbols to the total number of the received symbols. To further examine the impact of environmental dynamics on the performance of channel estimation, we repeat this experiment in three different scenarios, namely outdoor with the link distance of 2 m, indoor with the link distance of 40 m and outdoor with the link distance of 20 m.

Results: We plot the average symbol delivery ratio of the three scenarios in Fig. 9 when the frames are recorded as the JPG and RAW formats, respectively. The results show that four training symbols ($d = 15$) are not dense enough to eliminate the non-linearity effect, while performing piecewise interpolation with 16 training symbols ($d = 5$) can almost avoid symbol errors in the short-range scenarios, i.e., outdoor 2 m and indoor 40 m. We hence use 16 training symbols ($d = 5$) as our default setting for channel training and bilinear interpolation.

The case of indoor 40 m is more challenging and suffers from a higher error rate because the received intensity is already too low even when the longest exposure time is configured. However, our POLI can still ensure that the camera still receives more than half of the symbols correctly even in such a critical condition. The results show that, in general, RAW images produce a higher decoding success probability than JPG images, especially in the challenging long distance scenario. This explains why compression becomes a key factor that determines the final decoding performance when the received signal strength is not sufficiently high. Therefore, to support a very long range service, such as in a vehicular network, we might need to use a camera that is capable of outputting RAW images.

5.1.3 Impact of Rotation on Channel Correlation

The achievable spatial multiplexing gain of a multi-antenna system is typically determined by channel correlation among different received signals. For example, though a 3×3 MIMO system can theoretically support three simultaneous transmissions, the real throughput gain, however, cannot be triple if the channel vectors, i.e., $\mathbf{c}_1$, $\mathbf{c}_2$ and $\mathbf{c}_3$ in POLI, are not fully orthogonal with each other. In POLI, the channel vectors are determined by polarization rotation after the dispersion as well as the polarization direction at the receiver. Since a receiver might rotate to an arbitrary orientation, thereby changing its polarization direction, we hence check how varying channel correlation due to the receiver's rotation affects the symbol delivery rate.

Experiment setup: We place a receiving camera at 1 m away from the transmitter, which is sending random symbols of 32-Ary PIM signals. The exposure time of the receiver is set to 0.08 ms. We rotate the receiver to any orientation between 0 and 360 degrees. For each setting, the receiver recovers 100 symbols and measures its symbol delivery ratio for various rotations.

Result: We plot the symbol delivery ratios for various rotations in Fig. 10. The results confirm that POLI often achieves a symbol delivery ratio larger than 90%, showing that the color channels are mostly uncorrelated in most of the cases. The receiver might sometimes get bad luck to rotate to a polarization direction, e.g., rotating 80 degrees as shown in Fig. 10, that results in two color channel vectors $\mathbf{c}_1$ and $\mathbf{c}_2$ with a higher correlation, making it much harder to separate two simultaneous symbols. This problem is, however, unavoidable in most of the MIMO systems, including the RF communication protocols such as 802.11ac. In POLI, a possible way to remedy this situation is to replace the receiver's polarizer with a *liquid crystal cell* (LCC), i.e., a 1 pixel LCD, which can *electronically* adjust its polarization direction by charging it with different amounts of voltage. With the support of LCC, the receiver now should be able to search for a polarization direction maximizing channel orthogonality, and thereby the SNR of the received signals. We leave this as a flexible deployment choice.

5.1.4 Impact of Exposure Time on Service Range

The receiver's linearity range is closely related to its exposure time. We now check what is the optimal exposure time with respective to different distances of a light-camera link.

Experiment setup: In this experiment, we change the distance of the light-camera link, ranging from 1 m to 10 m. Both the transmitting light and the receiving camera are static. For each tested distance, we let the transmitter continuously send random 32-Ary

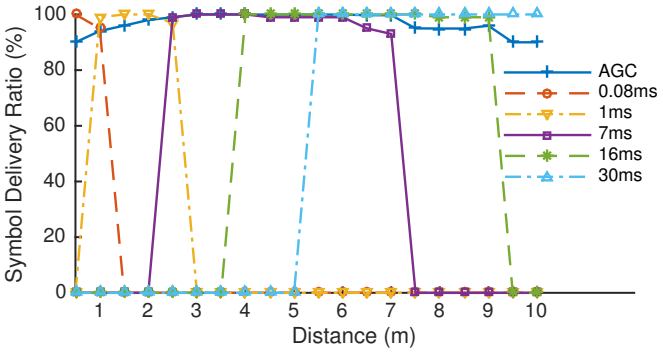


Figure 11: **Impact of exposure time on the service range.** Different exposure times have different operational ranges. A long distance link suffers from more severe fading and, hence, should configure a larger exposure time. With POLI's exposure time adaptation, we are able to quickly converge to a proper configuration and achieve a high delivery ratio at any distance.

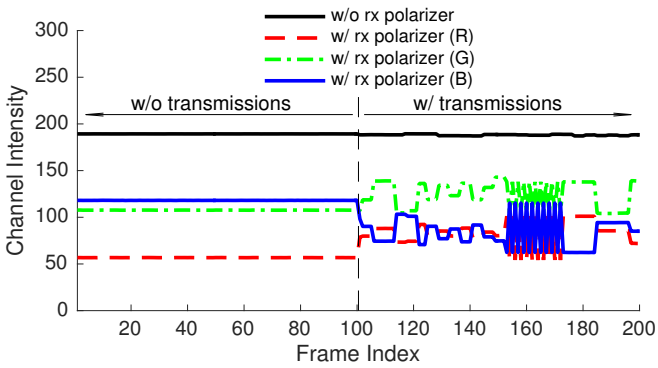


Figure 12: **Intensity variation.** POLI's PIM ensures that only a receiver attached with a polarizer can see intensity variation in its captured images, while the intensity variation of the aggregated light sent by the transmitter is very small and, thereby, indistinguishable by human eyes.

PIM symbols, while the receiver iteratively switches between all the five available exposure times. By this setup, the receiver could test different exposure times under a similar channel condition. The receiver then records 100 received symbols for every setting of the exposure times, and calculates the symbol delivery ratio of each exposure time, which is used to represent the performance when our AGC is disabled. For each tested distance, we then use the symbol traces of the five exposure time to offline emulate our automatic gain control (AGC), with the initial exposure time set to 7 ms and the thresholds $P_{\max}$ and $P_{\min}$ set to 80 and 150, respectively. The receiver then counts the number of correctly recovered symbols, during adaptation and after convergence, and calculates the achievable symbol delivery ratio when AGC is enabled.

Result: We plot in Fig. 11 the symbol delivery ratio of different exposure times (i.e., AGC disabled) and the symbol delivery ratio of our AGC, respectively, at different distances. The results verify that each exposure time has its own optimal operational range. Without proper configuration, the received intensity might be too dark or too bright, implying that the linearity range is too small to reliably detect all the possible symbol vectors. A camera usually receives a lower intensity at a long distance due to optical channel fading. Therefore, if a far receiver still configures a short exposure time, the received intensity will be too dark to be decodable. Similarly, a near receiver should configure a short exposure time to avoid overexposure. POLI adapts the exposure time to the receiver's channel condition, and can achieve a symbol delivery ratio close to the one achieved by the optimal exposure time. Since we only collect 100 symbols to evaluate the performance of AGC, the high delivery ratio also implies that our AGC can quickly converge to the optimal setting. Otherwise, if the convergence time is long, the symbol losses during adaptation would greatly degrade the overall delivery ratio.

5.1.5 User Perception

We finally evaluate whether POLI's polarization intensity modulation would cause flickers and become perceptible by human eyes.

Experiment setup: To numerically analyze how much illumination of POLI is perceived by human eyes, we remove the receiver's polarizer and measure the received aggregated light intensity sent by the three LEDs. We use the default settings listed in Table 2. The experiment is divided into two phases: *no-transmission phase* and *transmission phase*. In the *no-transmission* phase, each of the three LEDs continuously illuminate with a fixed randomly-selected intensity, but still ensures that the aggregated intensity equals to I_{total}, i.e., the constant total intensity used in POLI. In the *transmission* phase, the first two LEDs send random symbols, while the third LED illuminates the complementary intensity, following POLI's PIM design. Since human eyes (or a camera without a polarizer) still see the white light emitted by the transmitter, we use the average of the measured RGB values to represent the perceived light intensity. For comparison, we also allow the receiver to measure the received RGB values of the same signals with its polarizer installed.

Result: Fig. 12 illustrates that, after data transmissions, the change in the aggregated light intensity is negligible because POLI' PIM ensures that the total intensity keeps unchanged even when data are represented by intensity-based modulation. This figure further confirms that the variation of the RGB values is indeed large enough as expected, showing that our PIM can indeed modulate the *polarized color beams* with different intensity levels to increase the data rate. The figure demonstrates that our POLI successfully realizes an intensity-based modulation scheme, but introduces nearly no backlight intensity variation.

5.2 Throughput Performance

We now measure the overall network performance of POLI when the receiver is static but placed at different distances.

Experiment setup: In this experiment, we setup a light-to-camera link with a distance ranging from 1 m to 20 m. In each experiment, the transmitter sends 10 packets, each consists of a delimiter, the training symbols and 10 data symbols modulated by 32-PIM. The receiver updates its lookup table every packet and decodes the data symbols using the up-to-date table. The receiver applies POLI's automatic gain control and initializes its exposure time to 7 ms. Since some cameras on the market allow the user to manually configure the gain, we further check whether increasing the gain of a camera can extend the operational distance of POLI. Specifically, we further experiment when the camera locates at distance 20 m but increases its gain to 15 or 24. We test four different baud rates (the inverse of a symbol duration) to check how it affects the achievable service range. For each setting, we fix the location of the transmitter, but deploy the receiver at 5 randomly-selected locations at the specified distance and output the average result.

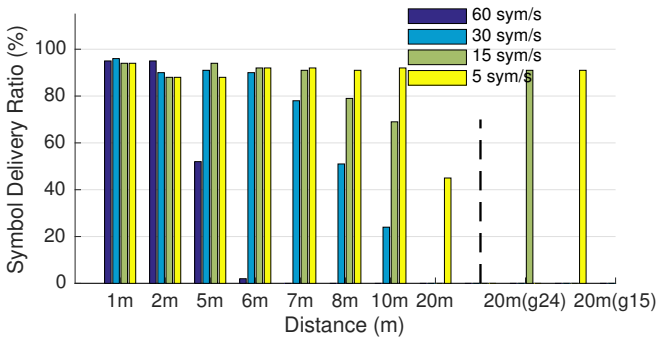


Figure 13: **Symbol delivery ratio at different distances.** The service range of different baud rates will be different. The maximum service range of a baud rate is bounded by the maximal feasible exposure time a receiver can use without suffering from the mixture frame problem.

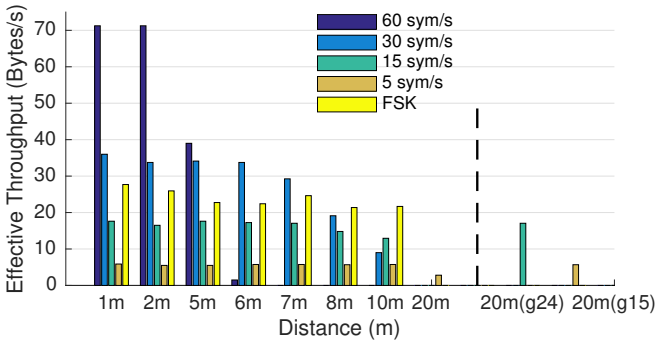


Figure 14: **Throughput at different distances.** A higher baud rate supports a shorter range but produce a higher throughput for a receiver within the service range. A transmitter can pick a baud rate to balance the tradeoff between the achievable throughput and the service range.

We further compare our POLI with conventional frequency modulation, i.e., frequency-shift keying (FSK) [18, 17]. The number of frequencies available for modulation is closely related to the link distance. We manually test the largest detectable frequency and smallest detectable frequency for each different link distance, and configure the frequency spacing as suggested in [18]. Intuitively, the more detectable frequencies we can use, the higher rate a receiver can get. For fair comparison, we similarly search the best exposure time setting, i.e., the one minimizing the symbol error rate, for FSK. Since FSK can only be applied in rolling-shutter cameras, we use a Pointgrey Flea3 camera with the same lens of POLI for FSK experiments. The output power of the transmitter, using only one LED, is configured to be the same as the total output power of the 3 LED lights of POLI.

Results: We plot the symbol delivery ratios and the effective throughput at various distances in Fig. 13 and Fig. 14, respectively. The figures show:

- The conventional FSK-based VLC scheme achieves a throughput comparable to our POLI at short distances. Its throughput, however, drops to zero when the distance is longer than 20m. The main reason is that the FSK-based scheme relies on a receiver being able to detect and estimate the width of dark and bright strips of a transmitting light from the captured images. However, at longer distances, light intensity decades significantly, and longer shutter time is required for the image sensor to collect sufficient light. However, in such cases the contrast of the strips becomes very small and hard to detect. Thus, the throughput of FSK eventually reduces to zero.
- Reducing the baud rate, i.e., increasing the symbol duration, extends POLI's service distance to 20 m since a receiver can avoid the symbol mixture problem even if it switches to a long exposure time. As a result, receivers can adaptively increase its exposure time to achieve a high enough signal strength for reliable decoding.
- As shown in Fig. 14, though a smaller symbol duration can only support a limited transmission range, its higher baud rate, however, corresponds to a higher effective throughput if a receiver is at shorter distance and has a high enough signal strength even when a small exposure time is used. The service range of this experiment is shorter than that shown in Fig. 9 since this experiment uses a higher-order modulation, i.e., 32-PIM, which needs a higher SNR to reliably decode the symbols. However, by increasing the camera gain, we can again achieve a fairly low error rate.
- We can observe from the two figures that the selection of a proper baud rate is actually very similar to the bit-rate adaptation problem [6, 33, 12, 22, 7, 20, 30], i.e., how to adapt the transmission bit-rate according to the SNR of a receiver, considered in 802.11 or a cellular system. By enabling a feedback channel to learn the channel condition of the worst receiver we want to serve, the transmitter can adaptively adjust its baud rate and make the best tradeoff between the achievable effective throughput and the service range. As this work mainly focuses on extending the communication distance, we leave the above baud rate adaptation problem as our future studies.

6. RELATED WORK

We can categorize recent related works into the following categories.

Light-to-camera communications: A number of literatures have focused on enabling light-to-camera communications. The systems [9, 23, 17, 18] leverages a high sampling rate of the rolling shutter effect of CMOS cameras to increase the data rate. However, these works all require the light sources to occupy a sufficient large number of rows in the captured image so as to capture the rolling shutter effect, which is, however, infeasible in along range scenario. Moreover, those systems need a receiving camera to operate at a low exposure time, which prevents the receiver at longer distances from getting sufficient intensity for reliable decoding. ColorBars [14] utilizes Color Shift Keying (CSK) to modulate data using different LED colors. However, the transmission range is very limited due to the high requirement on the received intensity. In [24], Luo *et al.* enable long range, low SNR communications through undersampled frequency shift OOK (UFSOOK). However, this approach does not utilize the dimensions spanned by the RGB space of a camera, as a result supporting a very low data rate, i.e., 0.5 bits per image. PIXEL [35] uses an LCD to modulate bits into the polarization directions of a light, and spans the polarized lights using a dispersor to ensure uniform intensity contrast among all the receiving directions. However, the transmission rate is limited by the low response time of an LCD.

Screen-to-camera communications: Recent works [5, 21, 13, 15, 34, 4, 16, 19, 31] modulate a huge number of pixels on a screen to boost the throughput of camera communication links. Strata [16]

implements the adaptive coding, supporting different transmission distances by adding layered coding to QR codes. COBRA [13] enhances the SNR of a screen-to-camera link by carrying additional color information in traditional barcodes. Further improvements include LightSync [15], which solves the inter-frame symbol problem caused by un-synchronization between a transmitter and a receiver, VRCodes (NewsFlash) [34], Inframe++ [31], ImplicitCode [27] and HiLight [19], which embeds extra information that could be extracted by a camera but not human eyes into the images. The work [4] proposes a rate adaptation scheme for camera-based communications. Benefiting from the high resolution of a screen, the above works all provide a high throughput. Nevertheless, this high resolution could hardly be captured by a distant camera. Also, the screen flickers as performing transmission may be obtrusive to human eyes.

Light-to-photodiode communications: Photodiodes features a high sampling rate and a high intensity resolution as compared to cameras. Many proposals [25, 32, 10, 28, 11] explore various application scenarios for the light-to-photodiode technology. LED-to-LED and OpenVLC [25, 32] use LED as the transceiver of a system, simplifying the hardware design of a communication system and enabling half-duplex communications. OWC [11] characterizes the optical channels of different optical modulations, and Li-Fi [10] provides numerous empirical data and boost the throughput so as to be comparable with an RF link. DarkLight [28] focuses on enabling communications with a low-duty cycle transmitter. The high intensity resolution of a photodiode provides a capability of of capturing signals far away; however, a photodiode sums all the incoming lights from different incidence angles, it usually takes a larger overhead to eliminate cross interference from multiple transmitters.

Bit-rate adaptation in WiFi: A number of literatures [6, 33, 12, 22, 7, 20, 30] have investigated the algorithms for bit-rate adaptation in 802.11. They either allow bit-rate selection at the transmitter side or receiver side, according to the dynamic channel condition of a transmission link. POLI's baud rate selection problem is similar to the conventional WiFi rate adaptation problem. However, unlike most of RF protocols supporting duplex communications, VLC usually supports an one-way light-to-camera communication. Fortunately, some recent research [25, 26] is dedicated to enable two-way visible light communications. We could expect that baud rate selection would be less a problem in the future with the support of duplex capability.

7. CONCLUSION AND FUTURE WORK

In this paper, we presented POLI, the first light-to-camera communication system that can serve camera receivers at long distance while the transmitter generates no flickers observable to human eyes. POLI uses a novel polarized light intensity modulation to allow the use of low frequency band in the signal, so that the data symbols can be transmitted at slower speed. As a result, without introducing inter-symbol interference, a camera receiver can use longer exposure time to accumulate sufficient energy, even at long distances, for reliable communications. POLI uses an optical rotatory dispersor to transform polarizations into color beams, and hence color cameras can take advantage of the three outputs of R, G, and B values to separate and demodulate the three transmitted data streams in different polarization directions, boosting the achievable data rate. POLI also incorporates a low-overhead channel estimation scheme and automatic gain control to deal with the channel non-linearity issue. We implemented POLI with off-the-shelf components and software defined radios (USRP). The evaluation results obtained from the prototype show that POLI produces throughput proportional to the channel condition, 34 bytes/s at 5 meters and 9 bytes/s at 10 meters, respectively, and can serve receivers at an extended range of 40 meters with an error rate of only 10%.

POLI makes an important step toward enabling long-range VLC. Some future works are worth investigated to further improve the reliability and applicability of POLI. First, in our experiments, we only report the raw loss rate and throughput. Therefore, though the loss rate at longer distances gets higher, we can combine POLI with proper error correction codes, such as FEC, to ensure low-rate reliable transmissions, thereby extending the service range of VLC. Second, while our trace-driven offline decoding evaluates the performance of POLI with real channels, we, however, do not deal with the synchronization issue between a transmitting light and a receiving camera, which could be a practical issue as implementing real-time decoding. Such a synchronization issue can be resolved by proper initial calibration, similar to the solution used in [18]. Finally, while the current proof-of-concept prototype uses off-the-shelf optical components, which has much larger dimensions than necessary, we believe that eventually with mass production the design can be shrank down to the size of a single LED chip, similar to the RGB LED chip available on today's market.

8. ACKNOWLEDGEMENT

This research was supported in part by the Ministry of Science and Technology of Taiwan (MOST 106-2633-E-002-001, MOST 104-2628-E-009-014-MY2), National Taiwan University (NTU 106R104045), Intel Corporation, and Delta Electronics.

9. REFERENCES

[1] Ettus Inc., universal software radio peripheral. http://ettus.com.

[2] Jackson Labs Technologies Inc., Fury GPSDO. http://www.jackson-labs.com/index.php/products/fury.

[3] Thorlabs Inc., non-polarizing beamsplitter cubes in 30 mm cage cubes. https://www.thorlabs.com.

[4] A. Ashok, M. Gruteser, N. Mandayam, T. Kwon, W. Yuan, M. Varga, and K. Dana. Rate adaptation in visual MIMO. In *IEEE SECON*, 2011.

[5] A. Ashok, M. Gruteser, N. Mandayam, J. Silva, M. Varga, and K. Dana. Challenge: Mobile optical networks through visual MIMO. In *ACM MobiCom*, 2010.

[6] J. Bicket. *Bit-rate Selection in Wireless Networks*. PhD thesis, Massachusetts Institute of Technology, 2005.

[7] J. Camp and E. Knightly. Modulation Rate Adaptation in Urban and Vehicular Environments: Cross-layer Implementation and Experimental Evaluation. In *MobiCom*, 2008.

[8] E. Collett. *Field Guide to Polarization*. SPIE Press, 2005.

[9] C. Danakis, M. Afgani, G. Povey, I. Underwood, and H. Haas. Using a CMOS camera sensor for visible light communication. In *IEEE Globecom*, 2012.

[10] S. Dimitrov and H. Haas. *Principles of LED Light Communications*. Principles of LED Light Communications: Towards Networked Li-Fi. Cambridge University Press, 2015.

[11] Z. Ghassemlooy, W. Popoola, and S. Rajbhandari. Optical wireless communications: System and channel modelling with MATLAB. Boca Raton, FL, USA, 2012. CRC Press, Inc.

[12] D. Halperin, W. Hu, A. Sheth, and D. Wetherall. Predictable 802.11 Packet Delivery from Wireless Channel Measurements. In *ACM SIGCOMM*, 2010.

[13] T. Hao, R. Zhou, and G. Xing. COBRA: Color barcode streaming for smartphone systems. In *ACM MobiSys*, 2012.

[14] P. Hu, P. H. Pathak, X. Feng, H. Fu, and P. Mohapatra. ColorBars: Increasing data rate of LED-to-camera communication using color shift keying. In *ACM CoNEXT*, 2015.

[15] W. Hu, H. Gu, and Q. Pu. LightSync: Unsynchronized visual communication over screen-camera links. In *ACM MobiCom*, 2013.

[16] W. Hu, J. Mao, Z. Huang, Y. Xue, J. She, K. Bian, and G. Shen. Strata: Layered coding for scalable visual communication. In *ACM MobiCom*, 2014.

[17] Y.-S. Kuo, P. Pannuto, K.-J. Hsiao, and P. Dutta. Luxapose: Indoor positioning with mobile phones and visible light. In *ACM MobiCom*, 2014.

[18] H.-Y. Lee, H.-M. Lin, Y.-L. Wei, H.-I. Wu, H.-M. Tsai, and K. C.-J. Lin. Rollinglight: Enabling line-of-sight light-to-camera communications. In *ACM MobiSys*, 2015.

[19] T. Li, C. An, X. Xiao, A. T. Campbell, and X. Zhou. Real-time screen-camera communication behind any scene. In *ACM MobiSys*, 2015.

[20] K. C.-J. Lin, N. Kushman, and D. Katabi. ZipTx: Harnessing Partial Packets in 802.11 Networks. In *ACM MobiCom*, 2008.

[21] S. D. Perli, N. Ahmed, and D. Katabi. Pixnet: Interference-free wireless links using LCD-camera pairs. In *ACM MobiCom*, 2010.

[22] H. Rahul, F. Edalat, D. Katabi, and C. Sodini. Frequency-Aware Rate Adaptation and MAC Protocols. In *ACM MOBICOM*, 2009.

[23] N. Rajagopal, P. Lazik, and A. Rowe. Visual light landmarks for mobile devices. In *IEEE IPSN*, 2014.

[24] R. D. Roberts. Undersampled frequency shift ON-OFF keying (UFSOOK) for camera communications (CamCom). In *IEEE Wireless and Optical Communication Conference*, 2013.

[25] S. Schmid, G. Corbellini, S. Mangold, and T. R. Gross. LED-to-LED visible light communication networks. In *ACM MobiHoc*, 2013.

[26] S. Schmid, T. Richner, S. Mangold, and T. R. Gross. EnLighting: An indoor visible light communication system based on networked light bulbs. In *IEEE SECON*, 2016.

[27] S. Shi, L. Chen, W. Hu, and M. Gruteser. Reading between lines: High-rate, non-intrusive visual codes within regular videos via ImplicitCode. In *ACM UbiComp*, 2015.

[28] Z. Tian, K. Wright, and X. Zhou. The DarkLight Rises: Visible light communication in the dark. In *ACM MobiCom*, 2016.

[29] D. Tse and P. Vishwanath. *Fundamentals of Wireless Communications*. Cambridge University Press, 2005.

[30] M. Vutukuru, H. Balakrishnan, and K. Jamieson. Cross-Layer Wireless Bit Rate Adaptation. In *ACM SIGCOMM*, 2009.

[31] A. Wang, Z. Li, C. Peng, G. Shen, G. Fang, and B. Zeng. Inframe++: Achieve simultaneous screen-human viewing and hidden screen-camera communication. In *ACM MobiSys*, 2015.

[32] Q. Wang, D. Giustiniano, and D. Puccinelli. OpenVLC: Software-defined visible light embedded networks. In *ACM MobiCom Workshop on Visible Light Communication Systems*, 2014.

[33] S. H. Y. Wong, H. Yang, S. Lu, and V. Bharghavan. Robust rate adaptation for 802.11 wireless networks. In *ACM MobiCom*, 2006.

[34] G. Woo, A. Lippman, and R. Raskar. Vrcodes: Unobtrusive and active visual codes for interaction by exploiting rolling shutter. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2012.

[35] Z. Yang, Z. Wang, J. Zhang, C. Huang, and Q. Zhang. Wearables can afford: Light-weight indoor positioning with visible light. In ACM MobiSys,2015.