



# SpectraLux: Towards Exploiting the Full Spectrum with Passive VLC

Seyed Keyarash Ghiasi\*

s.k.ghiasi@tudelft.nl

TU Delft

Delft, The Netherlands

Koen Langendoen

s.k.ghiasi@tudelft.nl

TU Delft

Delft, The Netherlands

Vivian Dsouza\*

viviand0097@gmail.com

TU Delft

Delft, The Netherlands

Marco A. Zúñiga Zamalloa

m.a.zunigazamalloa@tudelft.nl

TU Delft

Delft, The Netherlands

## ABSTRACT

In recent years, the number of wireless applications has increased significantly, resulting in the radio bands becoming expensive and prone to interference. There is a new research area aiming at mitigating these issues by creating communication links using *ambient light*. This area, called passive-VLC, not only exploits the visible light frequencies, but does so with low-power transmitters. All the previous work in passive-VLC, however, forget about individual wavelength bands of light, and do not exploit its wide spectrum, reducing the potential channel capacity. In this paper, we propose a novel method to transmit and decode data, using liquid crystal cells that modulate and consider the full spectrum, and put it to the test by prototyping a multi-symbol communication link. The main contribution of our work is to show that passive-VLC can move from *spectrum-agnostic* to *spectrum-aware* modulation. We explore this new domain by making use of a novel type of receiver (i.e., a spectrometer) and uncovering the advantages and caveats of this spectrum-aware approach.

### ACM Reference Format:

Seyed Keyarash Ghiasi, Vivian Dsouza, Koen Langendoen, and Marco A. Zúñiga Zamalloa. 2023. SpectraLux: Towards Exploiting the Full Spectrum with Passive VLC. In *The 22nd International Conference on Information Processing in Sensor Networks (IPSN '23), May 09–12, 2023, San Antonio, TX, USA*. ACM, New York, NY, USA, 14 pages. <https://doi.org/10.1145/3583120.3586966>

## 1 INTRODUCTION

Visible Light Communication (VLC) has emerged as a promising technology for wireless data transfer that escapes the crowded spectrum used by today's RF communication standards. VLC is envisioned to transform ubiquitous light sources in our modern society (lamps, screens, etc.) into a high-speed communication infrastructure that modulates information "for free" using the LED

\*These authors contributed equally to this work

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

IPSN '23, May 09–12, 2023, San Antonio, TX, USA

© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

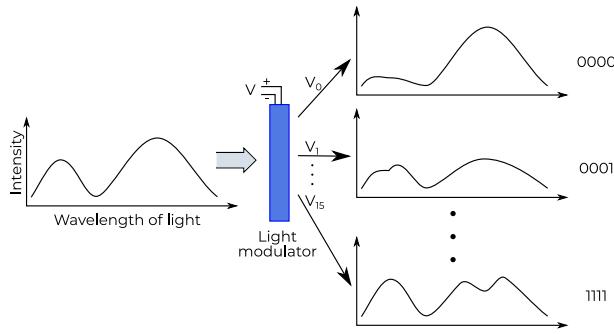
ACM ISBN 979-8-4007-0118-4/23/05...\$15.00

<https://doi.org/10.1145/3583120.3586966>

sources used for illumination. Recently –to support communication between low-power mobile devices– researchers have been developing *passive-VLC* systems in which ambient light (e.g., sunlight) is redirected by a "transmitting" device towards a receiving device; the ambient light serves as a carrier for data modulated on top of it. A common approach is to use liquid crystal (LC) shutters to block/let-through the incoming ambient light (e.g., on-off keying [12]); advanced schemes involve attenuating the light in a more fine-grained manner (e.g., amplitude modulation [18]). The use of ambient light saves several watts compared to active systems that include a power (illumination) source, and suits low-power IoT applications, for example communicating between a vehicle and road signs [16], between a surface and a phone camera [15], or for localization applications [14].

A key observation about the passive-VLC systems proposed so far is that they do not exploit the vast bandwidth (400 THz) included in the visible light spectrum (ranging from 375-780 nm). While RF communication, as well as fiber optical systems, make extensive use of sub-band modulation (e.g., OFDM), passive-VLC systems treat all incoming light as a single band, restricting themselves to amplitude- [5, 16, 20] and time-based modulation schemes [18]. Figure 1 shows an incoming spectrum on the left side. That spectrum is composed of many wavelengths that –in theory– could be modulated individually, which would allow for encoding many bits in a single light ray. This is important as commodity LC shutters have not been designed for switching at high speeds, and hence are the bottleneck in passive-VLC links. For example, the switching rate of some light shutters [2], is just 150 Hz, which severely limits the data rates that can be obtained with classic, *spectrum-agnostic* modulation techniques that typically reduce the whole n-dimensional spectrum into just one parameter (light intensity) observed with a photodiode. Camera-based or systems using a color-sensor operate with 3 (RGB) parameters, but still do not get close to the 256 sub-bands provided by the spectrometer.

We conjecture that the lack of spectrum-aware modulation in passive-VLC is due to a limited understanding of the physical properties of Liquid Crystal shutters deployed at the transmitter side. Yes, LC shutters can be toggled between translucent and opaque by switching the voltage from 0 V to 5 V, but what exactly happens during such a transition is not well-understood to most in the passive VLC community. Recent work by Ghiasi et al. has shed some light on the transition phase, allowing them to reduce the switching



**Figure 1: Main concept.** The spectrum of a light source (left) is controlled with a liquid crystal through changes in voltage. Communication symbols are mapped to different spectra (right). A receiver measures the spectrum to decode the data.

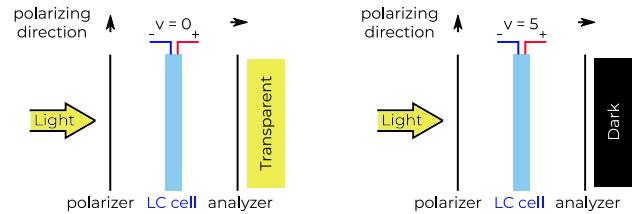
times by toggling between two intermediate *colored* states instead of pure opaque (black) and translucent (white) [10].

In this paper, we explore the possibilities of utilizing the broader spectrum provided by liquid crystal shutters commonly used for passive-VLC. The main challenge is that, in contrast to RF communication where the availability of band-pass filters with high Q factors enables transmitting and receiving individual sub-bands, LCs cannot be driven in such a fine-grained and orthogonal way. Yet, the voltage-driven spectrum emitted by LC shutters is rich and diverse, which allows for modulation schemes that go beyond the single dimension of light intensity, as depicted in Figure 1. Overall, our contributions can be summarized as follows:

- 1) *A theoretical model that describes how liquid crystals can be used to transmit different symbols in the spectrum domain.* By investigating the physical properties of LCs, in Section 2, we show that these devices can generate different spectrum patterns. Contrary to traditional modulation methods, where only the intensity of the whole spectrum is used to carry information, we increase the link's capacity by creating multiple symbols.
- 2) *A novel modulation-demodulation method that shows how our theoretical model can be used with off-the-shelf LCs.* In Section 3, we select symbols in the spectrum domain to transmit data. We will show that the received signals resemble spectrograms, and we decode symbols using machine learning algorithms.
- 3) *We build a prototype and evaluate it under different setups.* We build a transmitter based on LC cells and use a spectrometer in the receiver. To show the effect of the different light spectra, we test our link using different light sources, and compare the range and bit error rates (BER) of the system in various scenarios.

## 2 SPECTRUM-AWARE PASSIVE VLC

Our aim is to utilize as much of the visible light bandwidth as possible. When perceived by human eyes, different bands of light translate into colors. However, thinking in terms of color is too limiting; the visible light spectrum is much richer than the canonical 3-dimensional RGB color space. We will employ a spectrometer that captures 256 bands (15 nm wide) of the incoming light, which provides more room for multi-symbol coding *provided* that the transmitter can modulate these subbands. Multi-band modulation



**Figure 2: LC Operation.** The LC cell is placed between two perpendicular polarizing sheets. If  $v = 0\text{V}$ , the light is rotated and passes the analyzer; if  $v = 5\text{V}$ , the light is blocked.

is easy to accomplish with *active*-VLC, because we can control the light sources at a fine-grained manner, but we show that it can be done for *passive*-VLC as well despite the fact that the typical LC shutter has been manufactured to do just one thing well: switch between opaque (black) and translucent (white).

Recently, Ghiasi et al. [10] have shown that commercial-off-the-shelf shutters can be exploited to produce “colors” by (i) stacking layers of liquid crystals, and (ii) setting the driving voltages to intermediate values between 0 and 5 V. Ghiasi et al. used this insight to speed up OOK modulation by selecting two symbols that have slightly lower contrast (compared to methods using opaque/translucent states), but faster transition times. In this section, we deepen the understanding of the relation between input (driving voltage) and output (spectrum) of LC shutters as a stepping stone for the design and implementation of our novel platform, dubbed SpectraLux.

### 2.1 Basic functionality of LC cells

From a high-level perspective, LCs are transparent crystals that change the polarization angle of the light passing through. The rotation angle depends on a number of parameters including the voltage applied to the cell and the thickness of the crystalline material. LC cells are sandwiched between two polarizer films as illustrated in Figure 2. The first sheet, called the polarizer, ensures that the light entering the LC cell has a single polarization angle; the thickness of the cell has been determined such that when no voltage is applied ( $v=0$ ) the light is rotated effectively 90 degrees and can pass the second polarizer sheet, called the analyzer. The shutter is thus transparent by default. When a 5V voltage is put across, the crystal structure is changed such that the resulting rotation is zero causing the analyzer to block all light, making the shutter opaque.

Driving an LC cell with an intermediate voltage will result in reduced light intensity as well as a color shift. The latter aspect is important for designing a multi-symbol modulation scheme (Section 3). If we can change the color of the incoming light, we are –in effect– changing its spectrum, but exploiting this property requires a better understanding of the LC optics.

**2.1.1 Birefringence and path difference.** When light enters a transparent material, it travels at a lower speed compared to free space. The ratio between these two speeds is the material's refraction index. In LCs, the polarized incoming light breaks into two perpendicular waves (horizontal and vertical) traveling at their own speed, leading to two refraction indexes. These crystals are called birefringent materials. Birefringence is defined as  $\Delta n = n_l - n_h$ , where  $n_l$

$(n_h)$  is the refraction index of the lowest (highest) speed, respectively. Denoting  $d$  as the crystal's thickness, the *path difference* ( $\Gamma$ ) between the slowest and fastest waves is:

$$\Gamma = \Delta n \cdot d \quad (1)$$

When combined with the analyzer (cf. Figure 2), the path difference reduces the incoming light's intensity because only a fraction of the horizontal and vertical waves will pass the analyzer. According to [11], when the polarizer is perpendicular to the analyzer, the ratio  $L$  between the intensity of a wavelength after the analyzer, and before its entrance to the crystal, is given by:

$$L(\Gamma, \lambda) = \sin^2\left(\frac{\pi\Gamma}{\lambda}\right) \quad (2)$$

**2.1.2 The effect of different wavelengths.** Another key factor is the wavelength of the light passing through the LC. Until now, our analysis assumes that the light entering the LC is polarized and *monochromatic* (i.e. a single wavelength). When polarized *multi-chromatic (white)* light enters the crystal, each wavelength (color) will have a different path difference, and hence, a different attenuation  $L$ . The different level of attenuation faced by every wavelength is what allows us to change (modulate) the spectrum.

To compute the modulating effect on some specific ambient light source (e.g., a flashlight or sunlight), we need to account for the amplitude of the various wavelengths in its spectrum as follows:

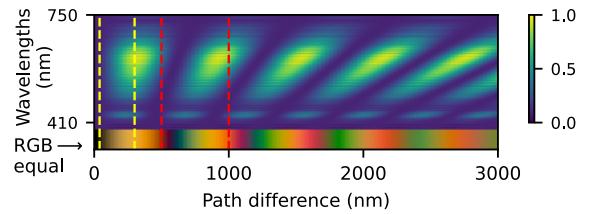
$$A_o(\Gamma, \lambda) = A_i(\lambda) \times L(\Gamma, \lambda) \quad (3)$$

where  $A_i$  is the amplitude of the input light for wavelength  $\lambda$ ,  $\Gamma$  the path difference of the LC cell, and  $A_o$  the amplitude of the light ray at the output of the analyzer. Taking all wavelengths into account will lead to the final color of the light exiting the LC shutter.

**2.1.3 Modifying the spectrum.** With the basic model captured in equations (1)-(3) we can study the impact of the path difference, which can be controlled by the LC voltage, on the spectrum emitted at the analyzer. This analysis is important as we need to select spectra that differ enough to serve as communication symbols. Figure 3 shows the effect of LCs (emulated with the above equations) on the spectrum of a phone's flashlight when increasing  $\Gamma$  from 0 to 3,000 nm. In the spectrogram, yellow denotes the maximum received intensity and dark blue denotes zero. To get a feeling of the spectrum, below the spectrogram we show the colors that would be perceived by the human eye using the CIE-1931 standard<sup>1</sup>. Once again, note that the human eye reduces the visible light spectrum to only three dimensions (RGB), and hence, color sensors or cameras provide only limited information.

Most off-the-shelf LC cells are made to operate between the darkest ( $\Gamma = 0$ ), and the first light region ( $\Gamma = 70$ ), shown by the dashed yellow lines in Figure 3. Although some gray and yellow tones can be obtained in between, richer spectra are present for higher path differences. Since the path difference can be increased with the crystal thickness  $d$  (Equation 1), stacking multiple LC cells and driving them with the same voltage, as depicted in Figure 4, allows reaching the richer spectra [10]. For example, if we stack two LCs together, we can cover the spectrum within the red dashed lines in Figure 3.

<sup>1</sup>The colors show a repetitive, though fading pattern, which can be explained from the basic operating principles. Due to space constraints, we cannot present that description.



**Figure 3: Modeled impact of path difference on flashlight spectrum per wavelength. The color on the bottom shows the combined effect of the wavelengths in the column above.**

## 2.2 Using LCs in communication

LCs have typically been used in passive-VLC by toggling them between transparent and opaque, which makes sense when using a photodiode as a receiver. With a spectrometer, however, we can select many more symbols (vertical lines in Figure 3) provided that their signatures (intensities at various wavelengths) differ enough. As stated before, stacking multiple cells increases the thickness (of the combined) crystal, enabling richer spectra. However, adding too many LCs has downsides. The first disadvantage is the attenuation of light caused by the glass panels used for each LC cell (discussed in section 7). A more severe problem is assuming that all LCs are switched at the same time with high precision, which is unlikely. As the number of LCs increases, the resulting spectra become less predictable or require long switching (stabilization) times. Consequently, the best approach is to use the fewest possible LCs that allows for sufficiently long inter-symbol distances.

To choose the optimum number of LCs, we need the  $\Gamma$  value, but most off-the-shelf manufacturers do not provide this parameter. Thus, for the pi cells used in this paper, which are manufactured by Liquid Crystal Technologies [1], we perform empirical measurements to obtain the effect of path differences on the spectrum patterns. We use the setup in Figure 4b, with a phone's flashlight as the input source, and a driving voltage sweeping from 0 to 20 V. This voltage is higher than the canonical 5 V used for driving typical LC shutters; in return, the pi cells switch faster: 400 vs. 150 Hz. Figure 5 shows the spectrogram for different numbers of LCs stacked together. When increasing the number of LCs, more patterns start appearing that resemble the trends of the theoretical model, cf. Figure 3. One noticeable difference is that, in reality, the changes in the spectrum start taking place at a voltage higher than 0 V; operating LCs below this voltage has no effect.

The main takeaway of our analysis is that it is indeed possible to use the driving voltage of LCs to control –to some degree– the output spectra. Based on our empirical results, we will be working with 4 pi cells from now on.

## 3 MODULATION

Modulation with LCs is more involved than just selecting a few symbols (columns) from the spectrogram in Figure 5; timing needs to be considered as well. LCs have a slow response, of a few hundred hertz. To provide enough time for the LCs to reach a steady state, in Figure 5, the measurements maintain the voltage level at the LC for 1 second. In a real link, the system cannot wait for such a long

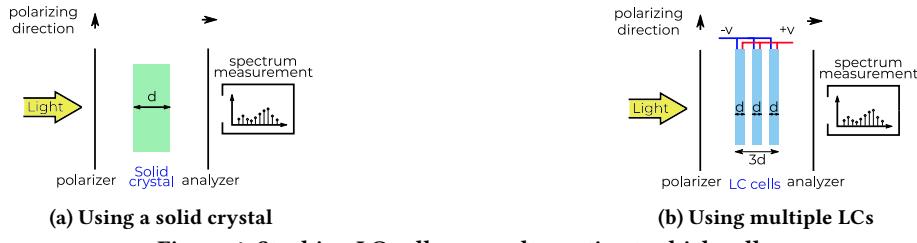


Figure 4: Stacking LC cells as an alternative to thick cells.

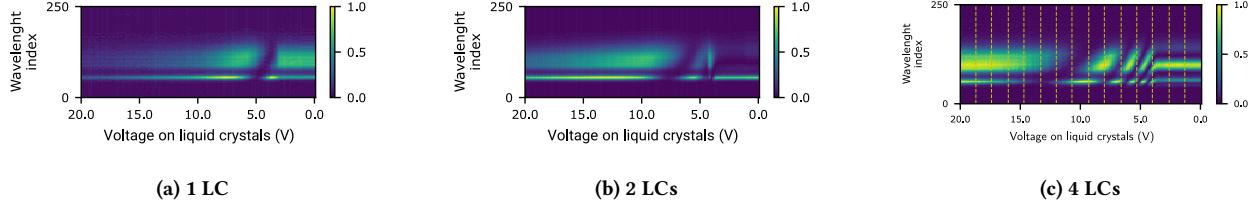


Figure 5: The effects of varying the driving voltage on the spectrum of a phone’s flashlight when passing through a stack of LCs. The voltages are plotted from high to low to match the corresponding theoretical results in Figure 3.

time, as the data rate would reduce dramatically. In this section, we provide a framework to increase the modulation speed.

### 3.1 Voltage analysis: borders vs. transitions

Following the characterization of the LC stack shown in Figure 5c, an intuitive approach to selecting communication symbols is to map each symbol to a unique voltage, as shown by the vertical yellow lines in the figure. But as stated before, we do not want to wait too long to reach a given voltage (symbol)<sup>2</sup>.

Instead of waiting to reach the desired symbol, we propose to use the transition between symbols. As can be seen in the spectrograms, switching between two voltages causes a continuous variance in the light spectrum. In our experiments, we found that the variation pattern is *unique* for each pair of start and end voltages. And with this transient pattern, we can predict the end voltage without waiting until it is reached. For example, if we have a transition between a starting voltage  $V_s$  and an end voltage  $V_e$ , we can infer that  $V_e$  is the intended symbol, simply by using the first part of the transition  $V_s \rightarrow V_e$ . We call the voltages *borders* to differentiate them from transitions. In section 6, we will show that this early decoding approach is valuable in reducing the symbol duration, and hence, increasing the data rate.

The trade-off for using transitions is higher decoding complexity. Instead of using a list of  $n$  voltages, the system needs to keep a list of  $n^2$  transitions, this process is explained in section 4. Next, we provide a deeper analysis of transition times in LCs.

### 3.2 Settling time

Considering that transitions occur between any pair of voltages, we are interested in identifying what range of voltages provides the fastest transitions.

<sup>2</sup>Following Shannon’s equation,  $C = B \times \log(1 + SNR)$ , which states that bandwidth increases the capacity linearly, as opposed to the SNR which only provides a logarithmic improvement, we prioritize reducing the symbol period (increasing the bandwidth) over maximizing the inter-symbol distance (increasing the SNR).

From a *materials’ science* perspective, some prior studies have looked into the dynamic behavior of pi cells. To measure how fast a cell can be switched, the *settling time* is defined as the time needed to achieve the optical transition when the cell is switched from a start ( $V_s$ ) to an end ( $V_e$ ) voltage. In [8], the dynamic behavior of pi cells is modeled using the hydrodynamic equations of Ericksen-Leslie. The associated simulations show a high variance in switching times between the same  $V_s$ , and two different  $V_e$  voltages.

From a *wireless communication* perspective, the high variability in switching times poses a key challenge: the symbol duration is not constant. Communication systems avoid variable-length symbols because the receiver is unable to use a fixed period to determine the boundary between symbols. To overcome this limitation, our approach follows two steps to define a *constant symbol period*:

- First, we measure the settling time in various regions of the spectrum in order to identify the region with the shortest switching times.
- Second, we select symbols in this ‘*fast-transitioning region*’ and define the symbol period as the slowest transition among these symbols.

Our two-step approach can be done with the help of simulations, however, to do so, one needs the physical properties of the pi cells, which are not provided by our supplier. Thus, we perform empirical measurements to analyze the fastest settling times.

**3.2.1 Settling time measurement.** To analyze the settling times, we select sixteen equidistant points (voltages) in the spectrogram defined for a stack of 4 LCs, as shown in Figure 5. After that, we measure the optical transitions between these voltages. Each transition is measured for 50 ms. The goal of this experiment is to find what region of the spectrum is the *fastest*, in terms of switching time between voltages. An example of a spectrogram transitioning between the 3rd (4 V) and 2nd (2.7 V) voltage levels is shown in Figure 6a. Notice that, after an initial *transitional* region, the spectrum remains rather constant. To calculate the settling time, a moving window is run along the horizontal axis, and the variance within

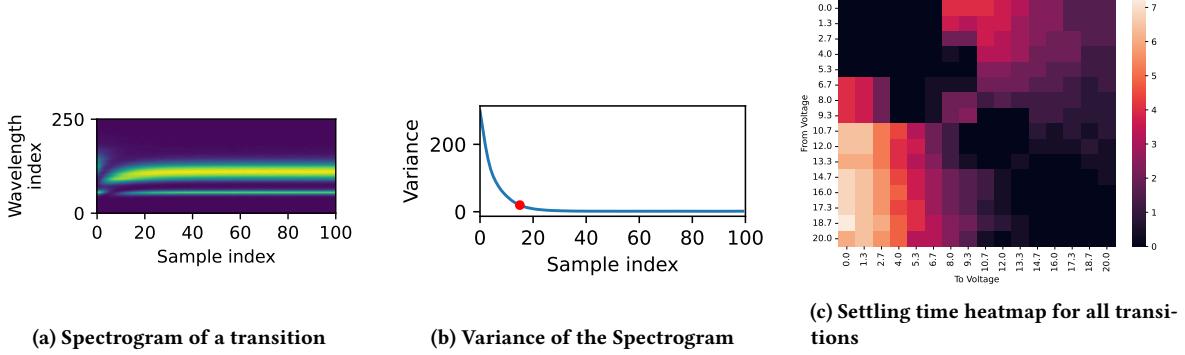


Figure 6: Settling time measurement steps and results.

the window is calculated for each wavelength (vertical axis). Then, we average the variance across all wavelengths for each vertical point, as depicted in Figure 6b. The settling time is defined as the point when the variance decays below a predetermined threshold, set to 20 in our case (red dot in Figure 6b). Following these steps, we generate the settling time matrix shown in Figure 6c, where the vertical axis represents the start voltage  $V_s$  and the horizontal axis represents the end voltage  $V_e$ .

**3.2.2 Analysis of the settling time matrix.** By looking at the matrix in Figure 6c, we get the following insights: *a) The matrix is not symmetric.* For many voltage pairs, going from high to low voltages takes longer than the reverse transition. The reason is that during a rising pulse, i.e.  $V_e > V_s$ , the liquid crystal changes under the influence of an electric field, whereas when  $V_e < V_s$ , the molecules rely on inertial forces to realign themselves. The force applied by the electric field is stronger than the inertial forces, and hence, rising pulses make faster transitions. *b) The settling time increases sharply when  $V_e$  is below the fourth voltage level* (equal to 5.3 volts). This occurs because the LC approaches its critical voltage  $V_c$ . This observation matches measurements done by Chen et. al. showing that when LCs operate around their critical voltage the relaxation times start to increase [7].

Overall, the settling time matrix indicates that symbols should be selected from the bottom right corner (darker region) because all transitions take  $\sim 2.5$  ms in both directions. Utilizing the entire matrix would entail considering settling times as high as 8 ms, reducing the data rate by a factor of  $\times 3.2$ . The trade-off for choosing this *fast* region is that we do not exploit the entire spectrum provided by the LCs. The fast, but narrower, region reduces the signal-to-noise-to-interference ratio (SNIR) and leads to a more condensed symbol constellation, i.e. the distance between symbols is shorter.

**Key design takeaway.** Given that the channel's capacity grows *linearly* with bandwidth (shorter symbol period) but only *logarithmically* with the SNR –as stated by the Shannon equation–, our design prioritizes bandwidth over SNR, and thus, we choose a *few fast symbols* (from the bottom-right part of the matrix) instead of selecting *many slow symbols* (from the entire matrix).

### 3.3 Constellations

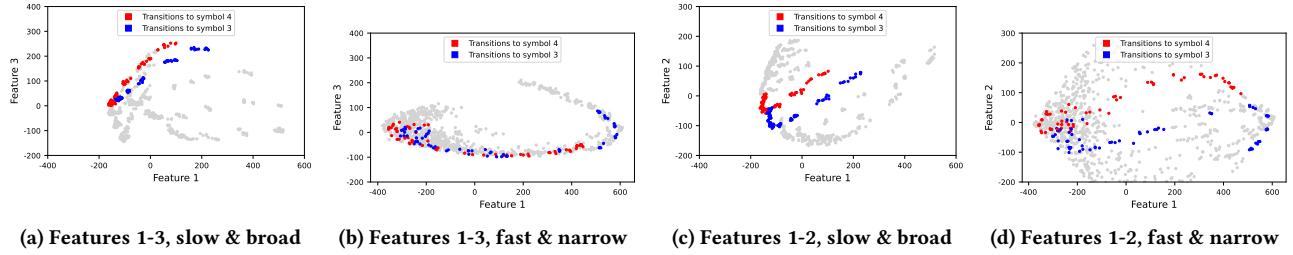
In the previous subsection, we mention that if all symbols (voltages) are selected from the darker region of the matrix in Figure 6c, the data rate can be higher than when symbols are spaced out over the entire range. This raises the concern that symbols might get too close leading to overlapping decision regions, which prevents their reliable detection. To investigate the relation between the overlap of decision regions and the width of the spectrum, we perform an experiment using the deployment presented in Figure 4b. The light source is a flashlight and the distance between the transmitter and receiver is 1 m. With that deployment, we consider two setups.

*Setup 1, broad spectrum:* Eight symbols are selected from the entire voltage range (0 to 20 V).

*Setup 2, narrow spectrum:* Eight symbols are selected only from the fast region in Figure 6c (10.7 to 20 V).

For each setup, we transmit one hundred messages containing all 64 transitions. Every transition is captured as a matrix, as shown in Figure 6a, where the rows are the wavelengths measured by the spectrometer and the columns are the number of samples. In section 4, we show that the information contained in the transition matrix can be compressed to 12 features while keeping more than 99 % of the information in the whole transition. In Figure 7, we use three of these features to give an idea of the effect of the spectrum width on symbol overlap. Figures 7a and 7c are from *Setup 1*, and figures 7b and 7d from *Setup 2*. Each dot is a transition and there are 64 transition clusters on each figure. Given that a symbol (end voltage) is a collection of eight transitions, those eight clusters belong to the same class. For clarity, the figures only highlight two contiguous symbol classes (red and blue) and leave the other six symbols in light gray.

The important observation from these plots is that reducing the spectrum width (left vs. right), reduces the distance between the symbols (as expected), which is most visible for the narrow spectrum in the top-right plot (Figure 7b), where the clusters of red and blue symbols largely overlap. Yet, even these seemingly “too narrow” and overlapping symbols can be decoded as other features (top vs. bottom) do separate them much more clearly as shown in Figures 7c and 7d, where the overlap is minimal for both spectra, broad and narrow. In general, the broader the spectrum, the more separation among the various PCA features, but since we will be



**Figure 7: Constellations of eight voltages (symbols) with the slow & broad spectrum (left) and the fast & narrow spectrum (right). In total, the system observes 64 transitions between voltages, and each symbol is represented by eight of those transitions. The features are obtained using Principal Component Analysis (PCA).**

using 12 PCA features to disentangle the symbols, there will be further dimensions to minimize the chance of complete overlap.

The point of this experiment is to provide an initial observation about the possibility of decoding symbols with a fast, but narrow spectrum. For a complete answer, we need to tackle two important points: (i) consider all transitions and features, and (ii) utilize a mechanism that can draw complex boundaries in the feature space to cope with the nonlinear patterns of our constellations. Next, we propose a demodulation method based on machine learning algorithms to tackle these two challenges.

## 4 DEMODULATION

At the receiver, a spectrometer captures the wavelengths of the light coming from the transmitter. A sample packet signal is shown at the top of Figure 8a. We can see that different patterns are generated when the voltage applied on the LC stack changes over time. The receiver captures the spectrogram as a  $w \times m$  matrix where  $w$  is the number of wavelengths at the receiver and  $m$  is the number of samples captured.

### 4.1 Symbol extraction

The communication link requires extracting the symbols from the spectrograms. To achieve this goal we perform three steps: parameter setting, preamble detection & clock recovery, and the actual symbol extraction as described below.

**4.1.1 Parameter setting.** A vital consideration in our approach is to minimize drift effects (accumulation of errors due to inexact measurements of the symbol period). If the transmitter's frequency and the receiver's sampling rate are not defined well, the number of samples per symbol varies. For example, the matrix of one symbol may have  $c$  columns and the next symbol may have  $c+1$  columns. These variations may seem small but they can cause severe detrimental effects in the machine learning process.

To avoid this problem, we make sure that the receiver captures an equal number of samples per symbol by setting the transmitter's symbol duration  $r_t$  as an *integer multiple* of the receiver's sampling period  $r_s$ :  $r_t = n \times r_s$ , where  $n \in \mathbb{N}$ .

**4.1.2 Preamble detection & clock recovery.** The receiver does not know the length of the transmitter's symbol period  $r_t$ , it only knows that it is set as an integer multiple of its sampling period. To identify the symbol's period, we design a preamble that causes

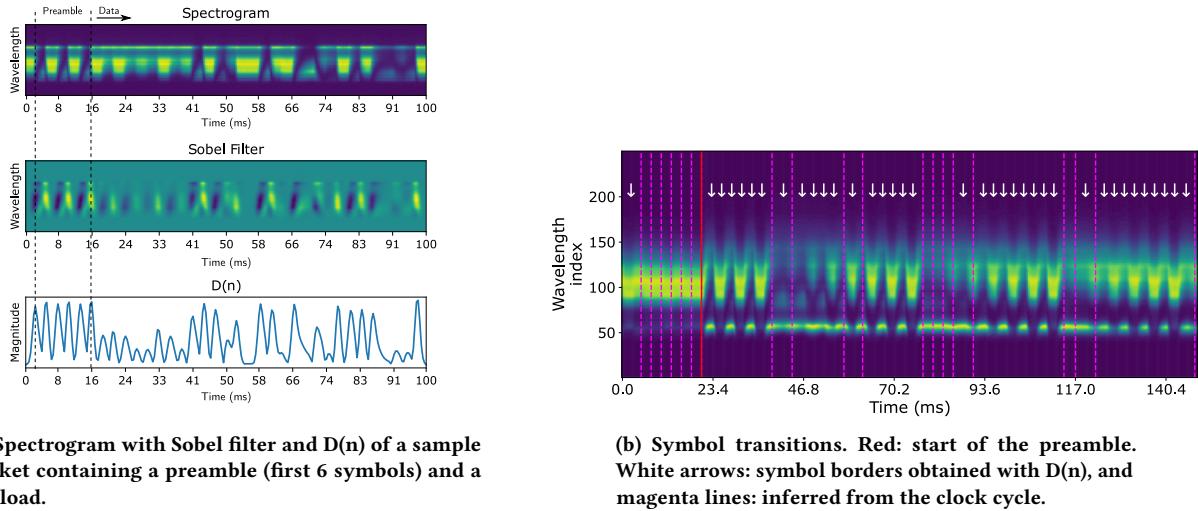
abrupt changes in the spectrogram, so it can be easily detected. The packet's preamble is designed to serve two purposes: a) recovering the transmitter's clock, i.e. identifying the value of  $n$  in  $r_t = n \times r_s$ , and b) marking the start of the payload. To detect the changes in the preamble, we consider the fact that the time response of LCs follows an exponential trend, similar to that of a capacitor: when the voltage switches from one symbol to another, the spectrogram changes with the highest gradient at the beginning, and the rate of change decreases as the LC reaches its final state.

To reveal the beginning of symbols, we run a Sobel filter along the horizontal axis of a spectrogram. The Sobel filter is a 1-D (row-wise) derivative in image processing used to measure gradients. The output of this filter is a matrix with the same size as the original image, as shown in the middle of Figure 8a. By summing this matrix along each column, we get the curve  $D(n)$  shown at the bottom of Figure 8a. The peaks in  $D(n)$  denote the times when the voltage on the LCs is switched, i.e. when the capacitance model exhibits the fastest change. To maximize the likelihood of detecting the peaks in the preamble, we use the two most-extreme symbols in the constellation. The transmitter sends 6 consecutive transitions between those two extreme symbols as shown by the first six peaks at the bottom of Figure 8a.

**4.1.3 Symbol extraction.** In theory, every transition between symbols should be detected in the Sobel plot. In practice, not all the beginnings of symbols generate a clear peak. This occurs mainly when similar or very close symbols are sent consecutively. Such an example is shown Figure 8a at around 54 ms where the payload contains two identical symbols back-to-back.

In this case, we estimate the beginning of the symbols based on the clock information obtained in the previous step. Whenever the distance between two peaks in the Sobel plot is greater than one clock cycle (which is extracted from the preamble), we divide that period into smaller components equal to the clock duration.

To test the validity of this method, we use a “helper wire” between the transmitter and receiver, signaling the start of packets and symbols. The estimated symbol borders turned out to be exactly the same as the ground truth. Figure 8b shows the results of our peak finding approach. The red line marks the beginning of a packet, the white arrows mark the symbol transitions detected by the Sobel plot (i.e. clear peaks), and the magenta dashed lines mark the transitions detected through the use of the inferred clock (i.e. unclear peaks).



**Figure 8: Symbol detection processing steps**

## 4.2 Feature extraction

With the borders found in the previous step, the spectrogram is split into smaller matrices (slices) for each symbol. The size of each matrix is  $w \times n$ , where  $w = 250$  (number of wavelengths) and  $n = 7$  (number of samples). The number of samples follows from the settling time and sampling rate of the spectrometer. Some examples of “transition slices” can be seen in Figure 8b.

The problem with these matrices is that they are big ( $250 \times 7 = 1750$  inputs), increasing the complexity of the overall system. To reduce this size, while retaining the most important information for each symbol, we apply a PCA algorithm as mentioned before. PCA dramatically reduces the dimension of the original dataset<sup>3</sup>. Figure 9a depicts the information captured by the most relevant components. We use the 12 most relevant dimensions which reduce the input size by more than  $\times 100$  and preserves more than 99 % of the original variance in the data. In our evaluation, we show that using 12 features provides links with a BER of 0 %. Using fewer than 12 features dramatically reduces performance, while only marginally improving computational overhead.

## 4.3 Classification: Training and Testing

Our system requires sending first training packets. These training packets contain all  $n^2$  transitions, so the receiver can obtain the required PCA features to decode the subsequent data packets. Figure 9b provides an overview of the entire communication process, including both training and actual packet transmissions.

There are various machine learning methods that can be used to classify the features obtained from the PCA results. As described in subsection 3.3, our symbol constellations require methods able to draw complex boundary conditions. After some preliminary investigations using different machine learning methods, we chose Neural Networks for this purpose, as they can adapt well to the needs of our demodulation process. Some Deep Learning techniques like Convolutional Neural Networks (CNNs) are also well suited to such

tasks. However, CNNs have a deep network with a large number of neurons requiring more memory and much more training data to avoid over-fitting. Therefore, in our approach, we use a simple fully connected neural network with 2 hidden layers of 64 and 32 neurons each, and an output layer of 8, or 16 neurons depending on the number of symbols chosen<sup>4</sup>. The model uses categorical cross-entropy for the loss function, and the Adam optimizer with a learning rate of 0.001. The training is carried out for 25 epochs.

For the link to work well, we need to provide the Neural Network with enough training data to successfully classify the  $n^2$  transitions. Exactly how much training data needs to be collected is not clear a priori and will be studied in section 6.

## 5 PLATFORMS

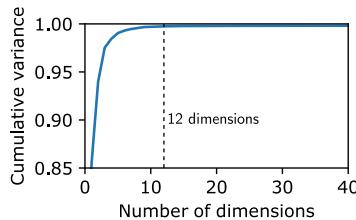
To evaluate the multi-symbol communication link, we build prototypes for the transmitter and receiver using commercially available devices. Next, we describe the systems, cost, and power analysis.

### 5.1 Transmitter

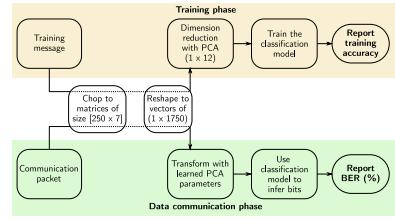
As mentioned before, in order to expand the spectral coverage, we stack 4 pi cells and connect their pins in parallel, as shown in Figure 4b. All these ports are actuated with a signal generator to create communication packets. Figure 10a shows our prototype transmitter. Non-polarized light enters through the polarizer, is modulated using the pi cells, and exits through the analyzer towards the receiver. The LCs we use could be modulated with very high voltages, up to 100 volts, however, due to the limits of the signal generator, the maximum actuation voltage that can be generated is 20 volts. Working at higher voltages has the advantage of increasing the modulation speed, and hence, increasing the data rate. At the same time, it consumes more power. Since our goal is to explore the use of the spectrum, using lower voltages does not change the proof-of-concept, and can be left as a design choice.

<sup>3</sup>The PCA implementation is obtained from scikit-learn [13].

<sup>4</sup>We use an implementation from Keras [9]



(a) Cumulative variance of PCA dimensions.



(b) Flowchart of the training and demodulation processes.

Figure 9: Cumulative variance as a function of PCA dimensions (a), and the system flowchart (b).

## 5.2 Receiver

The design of our receiver has to fulfill three requirements:

- (1) Measure the light spectrum with enough precision
- (2) Have at least twice the sampling speed of the communication link to satisfy the Nyquist requirement.
- (3) Be embeddable in a custom electronic system.

We considered three options: a low-end (AS7265X), a medium-level (Hamamatsu C12880), and a high-end (Thorlabs) spectrometer. The low-end option does not satisfy requirements (1) and (2) and the high-end does not satisfy requirement (3). For the final design, we choose the C12880 Hamamatsu spectrometer. This sensor operates as a 1-dimensional camera, where each pixel measures a small range of wavelengths, and all the wavelengths are read sequentially. This sensor has two important metrics, one is its pixel clock, and the other is the line (sampling) rate. The wavelength sensitivity of this sensor is from 380 nm to 850 nm, and the maximum line rate is 5 kilo lines per second. The prototype of our receiver is shown in Figure 10b. We design and build a driver PCB for the spectrometer, and record the intensity at each light wavelength. The driver circuit has the main task of generating clock pulses and converting the sensor values using an Analog-to-Digital converter (ADC). To focus the receiver on the area covered by the LC cells, we place a lens in front of the sensor. In this version of our receiver, the embedded microcontroller does not demodulate the information. The spectral data is forwarded to an external PC, which runs the classification algorithms and decodes the information.

## 5.3 Costs and power consumption

Cost and power consumption are two important factors when evaluating embedded systems, specifically for passive-VLC, where being low-power is key. In this section, we focus on these two parameters.

**5.3.1 Cost.** The table below shows the cost of the main components for the transmitter and receiver. As far as we know, this is the first work using a spectrometer as the receiver for passive VLC. To have a precise view of the phenomena, we chose the best integrated spectrometer we could find, however, with some engineering effort, accurate but cheaper receivers can also be made using analog spectrometers [3] and high-speed image sensors [4].

List of key components			
Component	Tx or Rx	count	Price per item
STM32 microcontroller	both	2	12 euros
Spectrometer	Rx	1	290 euros
Lens	Rx	1	5 to 25 euros
Pi cell	Tx	4	15 euros

**5.3.2 LCs' power consumption.** The speed of the transmitter is the key factor determining its power consumption. If an LC is modeled as a capacitor, its power consumption can be estimated by the following equation:

$$P = CV^2f$$

where C is the LC's capacitance, V is the voltage swing on the capacitor, and f is the frequency of the voltage V. Our measurements show that each pi cell has a capacitance of about 6.7 nF. Following the modulation method explained in Section 3, a 1 kHz pulse is put on 4 pi cells. With V peaking at 20 volts, the total power consumption is shown in the table below:

LCs' power consumption @ 1.3 kbps		
Component	Operating voltage	Power consumption
Pi cells	2 V to 20 V	0.7 mW

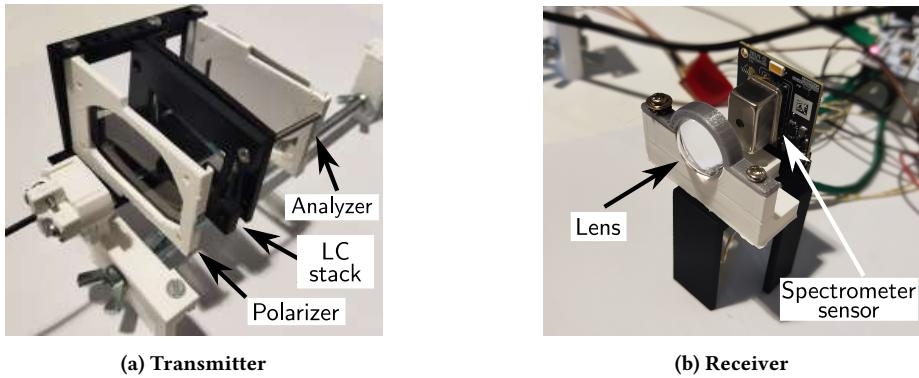
This sub-milliwatt consumption of LC cells is in agreement with the values reported by other passive VLC studies [10, 20].

**5.3.3 Spectrometer's power consumption.** In the receiver, the power consumption depends mainly on the clock frequency of the spectrometer. In our case, the spectrometer is clocked at 2.4 MHz. Following the information in the datasheet of the spectrometer, its power consumption is summarized below:

Spectrometer's power consumption		
Component	Operating voltage	Power consumption
C12880 Hamamatsu	5 V	0.166 mW

## 6 EVALUATION

In this section, we evaluate the performance of our system in terms of data rate, range, and bit error rate (BER). We explore various scenarios considering different numbers of symbols, ranges, light sources, and ambient lighting conditions. First, we evaluate one of the main insights of our study, which is that using *early* transitions to decode symbols is a plausible alternative, as opposed to waiting until the spectrum of the end voltage is reached. Then, we expose our system to different working conditions. And in the last part,



**Figure 10: Pictures of the transmitter and the receiver.**

we address the training cost of the machine learning models used for classification.

## 6.1 Borders versus transitions

During the demodulation, we use the initial part of the transition between two voltages (symbols) for the classification task. However, this increases the required training data quadratically. Due to this trade-off, we need to evaluate both methods to verify the advantage of using transitions. The following experiment reveals that when transitions are used, the minimum symbol duration with acceptable BERs is 2.6 ms. On the contrary, when using borders, the system needs to wait longer for a transition to settle: 3.4 ms.

Basic setup: To have a controlled environment, we set the experiments in a dark room with a light source placed directly behind the LCs. In this way, we minimize errors due to external sources and capture only the performance of the modulation technique. Additionally, the transmitter is kept within the line of sight of the receiver, with angle deviations of less than 10 degrees.

With this basic setup, we set the transmitter and receiver at a 1 m distance and send a constellation of  $n = 8$  symbols, so 3 bits/symbol. Each packet has 78 symbols, 6 symbols for the preamble, and 72 symbols to accommodate all 64 transitions. The transmission delay of each packet is between 0.2 s and 0.3 s, depending on the symbol period used, 2.6 ms or 3.4 ms, respectively. The sequence of transitions in the training packets is fixed and known, but to prevent over-fitting the neural network model, the data packets contain random sequences. For all the experiments done in this section (with different ranges, light sources and light intensities), we transmit 60 packets to train the Neural Network model, and then, 60 data packets to demodulate information.

Figures 11a and 11b show the results of the borders vs. transitions experiment. While the method using transitions achieves a BER below 1% for both symbol periods, the performance using borders degrades significantly when the symbol period decreases, increasing the BER from 0.9% to 4.2%. Considering that in passive-VLC the single-most-important bottleneck is data rate, many studies trade off complexity to increase the link’s speed. We follow the same principle, and hence, despite the heavier computational requirement, we use transitions for our system. In section 7, we discuss some possible solutions to reduce the training overhead.

## 6.2 Communication at different ranges

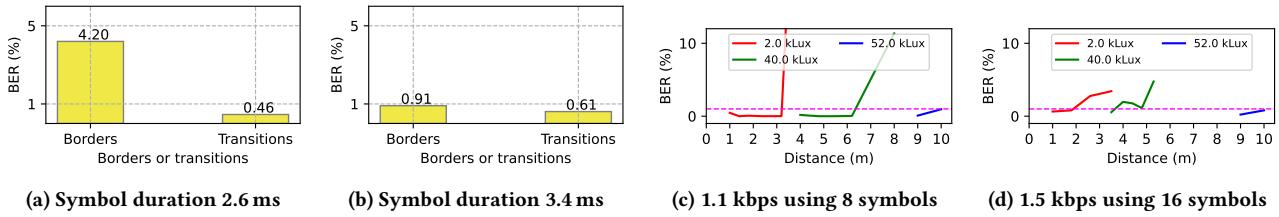
Now, we will compare the performance of SpectraLux at different ranges. For this purpose, we use the same *basic setup* as the prior section, but we change the distance from 1 to 10 meters. A picture of our test setup is shown in Figure 12a.

We test three different light intensities at the transmitter: 2 kLux, 40 kLux and 52 kLux. Each one of these light intensities has a minimum and a maximum range. The minimum range is due to saturation effects. Similar to photodiodes and cameras, spectrometers can saturate if the incoming luminous flux is too high. Hence, for each light intensity at the transmitter, we first identify the minimum range that prevents saturation, and then, we increase the range until the BER goes above 1%. In all the experiments we have two settings for the link's data rate: one with 8 symbols, leading to a data rate of 1.13 kbps, and the other with 16 symbols, leading to a data rate of 1.5 kbps. Following the analysis of subsection 6.1, the symbol period is always kept at 2.6 ms. The packet with  $n = 8$  symbols is the same as in the experiment in the previous part. The packet with  $n = 16$  symbols contains 278 transitions, including the preamble.

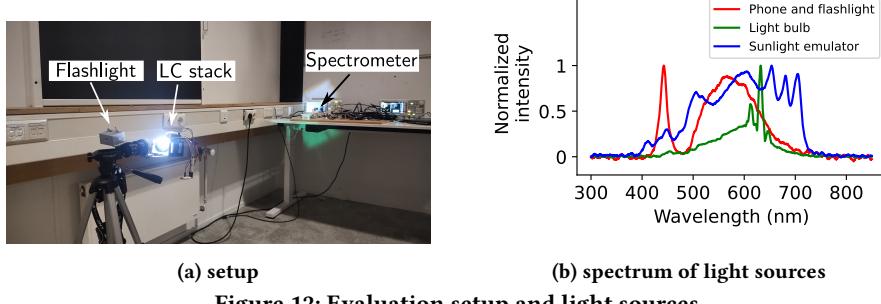
The results for the low and high data rate setups are shown in Figures 11c and 11d. In passive-VLC, an acceptable BER limit is 1%, which is shown by the horizontal magenta line. To put the light intensities in perspective, a 2 kLux is the illumination observed in well-lit indoor environments, while several tens of thousands lux are more similar to the illumination observed on a clear sunny day. Overall, these results indicate that SpectraLux can achieve acceptable BERs at a few meters range indoors.

### 6.3 Different light sources

In these experiments, we consider different light sources: a regular light bulb placed a few meters from the transmitter, and light coming from a sunlight emulator (a complex array of LEDs that emulate the spectrum of the sun). The normalized spectra of these light sources are shown in Figure 12b. In all previous experiments, a flashlight was used, whose spectrum is shown in red in Figure 12b. For these experiments, we only test the configuration with 8 symbols at 1.13 kbps, and consider three distances: 20 cm, 40 cm and 80 cm. The results are presented in the table below. The most important lesson of this evaluation is that the performance of the



**Figure 11: Comparison between using transitions versus borders (a) and (b), and BER performance at different distances (c) and (d).**



**Figure 12: Evaluation setup and light sources**

system not only depends on the strength of the light intensity (the higher the better) and the width of the spectrum (the wider the better), but also on the radiation pattern of the light source.

Bit Error Rate (%)					
Light source	Lux	Pattern	20cm	40cm	80cm
Phone flashlight	700	semi-direct	0.07	0.89	0.32
Light bulb	50	semi-direct	0.88	0.85	1.66
Sunlight emulator	10k	diffused	0.00	0.53	–

For any given light intensity, the more collimated (directional & parallel) the light rays are, the longer the range because the luminous flux is contained within a narrower beam [19]. For example, the phone flashlight has a high degree of directionality, and hence, it shows the best performance, with BERs below 1% for all ranges. The light bulb also has a high degree of directionality but due to the lower Lux, it has a higher BER at all distances. The sunlight emulator has the best performance at 20 cm (due to its wide spectrum) but the worst performance at 80 cm<sup>5</sup> because –in spite of providing the highest Lux among the tested light sources– the emulator provides diffused light, which scatters rays in all directions and only a small fraction is picked up by the spectrometer. The same phenomenon is reported in [19], which states that diffused light is the least ideal condition for their passive-VLC platform. With real sunlight on a sunny day, the rays would be highly directional, and hence, we hypothesize that if, similarly to the setup used in [10], we place a mirror at the transmitter (to direct sunlight in a specific direction), the range increases.

<sup>5</sup>There was no signal present in the link due the diffusion of light.

## 6.4 Variation in light intensity

In all previous experiments, the light intensity is kept constant during the training and demodulating phases. When using ambient light indoors, we can assume that the intensity does not change significantly, so both the training and communication phases are carried out using the same illumination. *However, this assumption does not hold true in all scenarios.* Under varying lighting conditions, an important question is to assess if the training done with one light intensity is valid to demodulate data at a different light intensity. To test this effect, we select five decreasing intensity levels, labeling them from L1 to L5, and experiment with three different scenarios:

- 1) We train our model using a combination of L1 and L5, and test it on all light intensities.
- 2) We repeat experiment (1), but train our model only with the highest intensity (L1).
- 3) We repeat experiment (1), but train our model only with the lowest intensity (L5).

The result of these experiments is shown in Figure 13a. The main outcome of this evaluation is that the training and testing phases have to be done with the same light intensity. When training is done with L1 (red curve), only the testing done at that light intensity has a low BER. When both, L1 and L5 are used for training (blue curve), only those two intensities provide a low BER. This indicates that under different Lux, the wavelengths in the spectrum change in a non-linear manner. Thus, in order to have a reliable link, *the training has to be redone whenever the received light intensity changes.* Hence, we analyze the cost of training the model in the next part.

## 6.5 Training overhead

Until now, we have been using 60 packets to train our model. However, due to the need for retraining, as a result of changing light

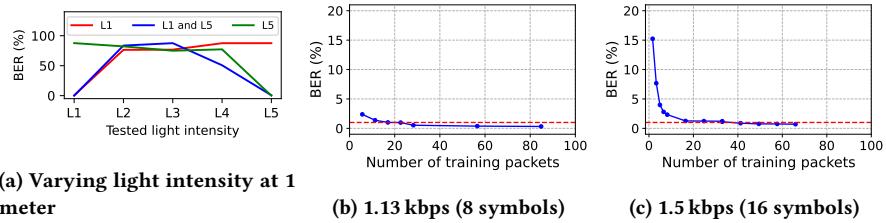


Figure 13: Amount of data required to train the ML model.

intensities, it is important to identify the minimum number of training packets required to achieve a low BER during the demodulation process. The effect of the amount of training data on the BER can be seen in Figure 12. In this experiment, we transmit at a distance of 20 cm with the *basic setup*. The amount of training required for each constellation is summarized in the following table.

Training time for BER <1%		
Data rate	Time or Packets	Neural Network
1.1 kbps	Packets required	17
	Time required (s)	3.4
1.5 kbps	Packets required	37
	Time required (s)	27.2

We observe that the convergence time varies with the constellation size (number of symbols). For eight symbols, the system requires 3.4 seconds to transmit 17 training packets, while for 16 symbols, the number of packets and time increase to 37 and 27.2 s respectively. Overall, indoors, the amount of training would not be a major issue because ambient light remains relatively constant.

## 7 DISCUSSION

To the best of our knowledge, this work is the first to exploit the spectrum of ambient light for passive communication. By and large, the main challenge in passive-VLC is to increase the data rate. Due to the slow response of LCs, most SoA studies improve the link speed by increasing the complexity of the hardware and/or the modulation method, but while still using the entire spectrum in a monolithic manner. Our approach provides a new perspective to exploit the broad light spectrum. Except for RetroTurbo [18], which uses a large surface with 64 LCs (*multi-pixel* transmitter) and an advanced QAM technique to achieve 8 kbps, all the other *single-pixel* transmitters [5, 10, 12, 16] reach a maximum data rate around 1 kbps. SpectraLux moves the needle for data rates with single-pixel transmitters, but there are several issues to resolve, as discussed next.

### 7.1 Methodology

SpectraLux's methodology has two important points that need to be improved: reducing the complexity of the channel estimation (training phase) and optimizing the symbol selection.

*Channel estimation.* As described in subsection 4.3, the amount of time required for channel estimation grows quadratically with the number of symbols because if we have  $n$  borders (symbols), we need to send  $n^2$  transitions during the training phase. If those

transitions could be generated via simulations, the training phase would not have to be done in real-time.

To achieve an ‘emulated’ training system, we want to combine the *static* spectrogram derived in Section 2, and the timing properties of the capacitance model, discussed in Section 3. The spectral characterization shown in Figure 5 captures the portion of the spectrum covered between any two symbols (transitions), but it does not consider that the capacitance effect warps this signal, as depicted in Figures 6a and 6b. The capacitance model, on the other hand, captures the speed at which this portion is traversed (initially fast and later slow). By combining the static spectrogram and the dynamic capacitance model, we could emulate the required transitions and eliminate the need for transmitting the training packets. The only parameters that we would need are the spectrum of the light source, and the dynamic model of the LC.

We tried this hypothesis with slower types of LCs, called Twisted Nematic (TN) cells, using a second-order constant capacitance model. Figure 14a shows the results of this emulation along with the real measurement of the symbol. Visually, the real and empirical transitions look similar, but when SpectraLux decodes packets using the emulated training, the BER increases from less than 2% to 83% (for 8 symbols). This occurs because the constant capacitive approach is not a precise way of modeling LCs. As mentioned in [7, 8], there are more accurate differential equations to describe the transient behavior of LCs. An important next step is to build upon those models to create more precise representations of the transitions.

*Symbol selection.* In section 3, the *selection of symbols is equidistant* due to the piece-wise linear trends present in parts of the spectrogram. This heuristic, however, is not guaranteed to be optimal. Obtaining the optimal symbols would require an exhaustive search over the entire spectrogram, but we are investigating alternatives to attain near-optimal solutions. Figure 14b depicts a symbol selection process using an unsupervised K-means clustering algorithm. This approach automatically clusters all the columns in the spectrogram into sixteen different classes and selects the centroid of each class as the representative symbol. We found out, however, that this particular alternative results in a much lower accuracy. For a 16-symbol experiment, the equidistant method provides 98% accuracy, while the clustering approach provides 8%. The poorer performance is due to the fact that the clustering approach does not take the dynamic LC behavior into account.

Overall, we believe that the research community working on passive VLC requires a deeper collaboration with material scientists to understand the transient state of LC cells. Several studies,

including ours, exploit in one way or another the transient state of LC cells [5, 10, 16, 20], but without considering in detail the mathematical models and properties reported by the LC research community.

## 7.2 Transmitter

Our transmitter requires placing multiple cells in series. This design causes a high level of attenuation that could be avoided if we could design our own cells. To cover a wide spectrum, we need specific values for thickness and birefringence for the LC layer, and this layer would need to be placed between two *glass* electrodes. In our case, we attain the required thickness and birefringence by placing multiple cells together. This means that, instead of using a single pair of glass electrodes, our design has multiple glass panels that attenuate (unnecessarily) the signal intensity. After measuring the attenuation of our LC cells, including the glass and crystal layers, we saw that four cells reduce the light intensity by a factor of 4.4, compared to a factor of 1.1 with a single cell. A custom-made cell would eliminate the attenuation of the redundant glass panels. In addition to decreasing the attenuation, designing a cell enables the transmitter to include a more diverse spectrum, allowing for more symbols and an increase in the SNR.

Another important question is if the communication is affected by slight manufacturing variations among the same model of LC cells. To investigate this, we use 2 different stacks of pi-cells in addition to what was used in Section 6. We repeat the spectrum characterization and the communication experiments at 1 m using all three stacks. The results of the characterization are shown in Figure 15, and except for the slight variations in the unstable region of low voltages, the patterns look the same. In addition, the BER in all these experiments remained consistent at 0%. This resilience occurs because, even if there are slight manufacturing differences among stacks, they can be accounted for during the training period. Thus, replacing LCs of the same model does not pose any problems to the link.

## 7.3 Receiver requirements

In this paper, the training and demodulation are done on a PC. In case a fully embedded receiver is required, the right microcontroller has to be chosen. The microcontroller should have sufficient clock frequency and RAM to run the decoding operations. Regarding the former, a 2 Mhz clock is needed for the spectrometer, as well as an ADC capable of sampling at this rate. This condition is satisfied by most microcontrollers. However, the bottleneck for the demodulation is the available RAM. To determine the memory requirements of the testing (decoding) phase<sup>6</sup>, we list the operations in the following table, assuming that all variables are 8-bit integers.

We can observe that at least 46 kB of RAM is needed for the demodulation pipeline, without including the code, libraries, and middleware overhead. This amount of memory is available only in middle- to high-end microcontrollers. Hence, it will be important to choose the right low-power microcontroller considering memory limitations or to allow for the addition of off-chip RAM to the receiver's circuitry.

<sup>6</sup>The training phase is done offline

Operation	Memory (lower-bound)	Size(B)
Input stream	250 × 7 (pixels) × 6 (symbols)	10500
Sobel filter output	Same as input stream	10500
D(n) signal	6 (symbols) × 7 (samples)	42
PCA matrix	250 × 7 (pixels) × 12 (features)	21000
NN (input to layer 1)	12 × 64 (matrix) + 64 (bias)	832
NN (layer 1 to layer 2)	64 × 32 (matrix) + 32 (bias)	2080
NN (layer 2 to output)	32 × 8 (matrix) + 8 (bias)	264
Total		45218

## 8 RELATED WORK

The analysis of the state-of-the-art (SoA) focuses on platforms using liquid crystals for *passive* communication. These studies use different types of liquid crystals, ambient light and surface areas, and hence, achieving an apples-to-apples comparison is not simple. In spite of this wide range of alternatives, our work provides a key contribution: to be the first to modulate the spectrum. For the purposes of our work, we divide the most relevant studies into two main groups, reflective and transmissive, and summarize them in a table at the end of this section.

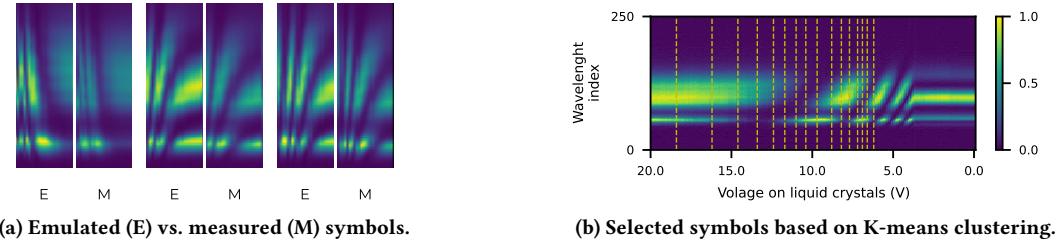
### 8.1 Reflective

The idea of using (retro)reflectors for communication was first proposed in smart dust via mechanical oscillations [17]. More recently, LCs were proposed to avoid the need for mechanical movements. The first study in this area, Retro-VLC, uses on-off keying with Manchester encoding. With this approach, the LCs use only two states: transparent and opaque. Depending on the type of LC used, the data rates range between 250 bps [20] and 500 bps [12]. Motivated by this work, subsequent studies propose ways to overcome the slow discharging phase of LCs. The common theme of all these studies, however, is to focus on intensity, not on the spectrum.

Passive-VLC proposes trend-based modulation [20]. Instead of exploiting only the end states (opaque and transparent), the system tracks the slope of the slow (discharging) signal. This approach, combined with Miller coding, achieves a data rate of 1 kbps. RetroI2V [16] uses the same modulation scheme but modifies three parameters to increase the range: a bigger surface (36×), a 30 W lamp (instead of the 3W lamp used in Passive-VLC), and two photodiodes to improve the SNR. With these modifications, RetroI2V achieves the same data rate as Passive-VLC (1 kbps), but the range is increased from 1 m to 80 m.

Instead of trying to tame the slow discharging phase, RetroTurbo proposes delayed superimposed modulation to exploit the *faster charging phase* [18]. Superimposed modulation, however, is constrained by the type of LC. For example, if we use the same LCs as in PassiveVLC, instead of increasing, the data rate would actually be cut by half. To overcome this limitation, RetroTurbo uses a liquid crystal where the charging phase is ten times faster than the discharging phase. Furthermore, RetroTurbo uses a QAM constellation with 16 symbols. The QAM method requires 64 LC cells at the transmitter and four photodiodes at the receiver, but together with superimposed modulation achieve an impressive 8 kbps.

PassiveVLC, RetroI2V and RetroTurbo are notable contributions, but their approaches introduce two challenges. First, inter-symbol dependence. With trend-based and superimposition modulation,



**Figure 14:** Potential methods of (a) reducing the training overhead using symbol emulation, and (b) optimizing the symbol selection process.



**Figure 15:** Characterization of different stacks of LCs. Apart from the unstable region in low voltages, the patterns are the same.

Comparison with the state-of-the-art (SoA)						
Name	Light source	Data rate	Range	Tx Power (only LCs)	Tx LC count	Receiver
RetroVLC[12]	LED lamp	0.5 kbps	2.4 m	0.234 mW	1	Photodiode
PassiveVLC[20]	LED lamp	1 kbps	1 m	0.525 mW	1	Photodiode
PIXEL[21]	LED/Ambient light	14 bps	10 m	1 mW	1	Camera
RetroTurbo[18]	LED lamp	8 kbps	7.5 m	0.8 mW	64	Photodiode
LuxLink[5]	Ambient light	80 bps	4.5 to 65 m	0.4 mW	1	Photodiode
ChromaLux[10]	Ambient light	1.1 kbps	1 to 50 m	0.166 mW	4 to 6 (series)	Color sensor
SpectraLux	LED lamp	1.5 kbps	10 m	0.7 mW	4 (series)	Spectrometer

symbols are not independent; the waveform of a symbol is affected by prior symbol(s). Second, reduced resilience to noise. By dividing the total amplitude level into sub-levels, the system becomes more prone to errors due to changes in the surrounding ambient light. Our work proposes using a more complex receiver (spectrometer) to modulate the spectrum, instead of intensity, without creating inter-symbol dependence and removing the constraints of *single* intensity modulation.

In addition to LC-based reflective systems, some recent platforms are exploring the use of digital-micro-mirror devices (DMDs) to modulate ambient light [19]. Compared to LCs, DMDs have a much higher switching speed, enabling them to achieve data rates of up to 80 kbps. However, the communication beams are narrow (creating strict alignment requirements) because the area of DMDs is just a few  $mm^2$ , compared to the much bigger  $cm^2$  area of LCs.

## 8.2 Transmissive

Backscattering systems based on LCs require artificial light sources, and the receiver needs to be co-located with that light. Transmissive platforms can work with any type of ambient light and allow placing the receiver at different locations. Transmissive systems

fall into two main categories based on the receiver type: photodiodes or color sensors. Our platform is also transmissive but uses a spectrometer.

**Black-and-white.** Contrary to backscattering studies, which rely on amplitude based modulation, LuxLink [5] proposes the use of frequency-shift-keying to cope with the stochastic behavior of sunlight. The system achieves long ranges (65 m) but at a low data rate (80 bps). Luxlink provides a novel sunlight link, but it still relies on monochromatic changes of light.

**Color-based.** A few studies use liquid crystals to generate color-based patterns. Pixel [21] and Poli [6] add a dispensor to an LC to move towards the color spectrum, and use color shift keying to transmit data to a camera. This approach achieves resilience for tag rotations, but does not exploit the transient state of LCs. They only use the end-states, and thus, achieve a low data rate: 14 bps for Pixel (using a single shutter) and 72 bps for Poli (using three shutters with three LEDs). In contrast, ChromaLux exposes the novel transient state in LCs to modulate a single color channel at fast speeds [10]. That approach allows exploiting sunlight to achieve a long range (50 m) and high data rate (1.1 kbps). Our work is inspired by these studies, but contrary to them we perform a thorough analysis of the

transient state of LCs to modulate a broader portion of the light's spectrum.

## 9 CONCLUSION

In this work, we explored an overlooked property of liquid crystal cells used in passive VLC systems: the idea that these devices can be used as controllable light filters in the spectrum domain. We present a theoretical framework to analyze this property and propose modulation and demodulation methods to create a multi-symbol communication link. By means of exploiting the transitions between symbols, instead of using the symbols themselves, we manage to surpass basic intensity modulation relying on a machine learning approach. Yet limited, this work sheds some light on methods to control ambient light wavelengths, and introduces a novel alternative to the passive VLC community, paving the way towards a more efficient use of the ultra-wide bandwidth of the visible light spectrum with available off-the-shelf devices.

## ACKNOWLEDGMENTS

This work is part of the *LuxSenz* project, a *TOP-Grant, Module 1, Physical Sciences* with project number 612.001.854, which is financed by the Dutch Research Council (NWO).

## REFERENCES

- [1] 2016. Liquid crystal technologies. <http://liquidcrystaltechnologies.com/>. Accessed: 2022-05-12.
- [2] 2020. Active 3D glass. <https://bit.ly/31hKJ4m>.
- [3] 2022. Benchtop Spectroscope. <https://www.patonhawksley.com/product-page/benchtop-spectroscope>.
- [4] 2023. hamamatsu line sensors. <https://www.hamamatsu.com/us/en/product/optical-sensors/image-sensor/ccd-cmos-nmos-image-sensor/line-sensor/for-industry.html>. Accessed: 2023-02-15.
- [5] Rens Bloom, et al. 2019. LuxLink: creating a wireless link from ambient light. In *Proceedings of the 17th Conference on Embedded Networked Sensor Systems*. 166–178.
- [6] Chun-Ling Chan, et al. 2017. POLI: Long-Range Visible Light Communications Using Polarized Light Intensity Modulation. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services* (Niagara Falls, New York, USA) (*MobiSys '17*). ACM, New York, NY, USA, 109–120.
- [7] Hongfei Cheng and Hongjin Gao. 2001. Dynamic simulation of Pi-cell liquid crystal displays. *Liquid Crystals* 28, 9 (2001), 1337–1341.
- [8] Shu-Hsia Chen, et al. 2002. Dynamics of twisted nematic liquid crystal pi-cells. *Applied physics letters* 80, 20 (2002), 3721–3723.
- [9] Francois Chollet, et al. 2015. Keras. <https://keras.io>.
- [10] Seyed Keyarash Ghiasi, et al. 2021. A principled design for passive light communication. In *Proceedings of the 27th Annual International Conference on Mobile Computing and Networking* (New Orleans Louisiana). ACM, New York, NY, USA.
- [11] R. Kanaris-Sotiriou. 2000. Bloss, F.D. Optical Crystallography.: Mineralogical Society of America Monograph Series, Publication #5, 1999. *Mineralogical Magazine* 64, 4 (08 2000), 777–778. arXiv:<https://pubs.geoscienceworld.org/minmag/article-pdf/64/4/777/2912366/777.pdf>
- [12] Jiangtao Li, et al. 2015. Retro-VLC: Enabling Battery-free Duplex Visible Light Communication for Mobile and IoT Applications. In *Hotmobile*.
- [13] F. Pedregosa, et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
- [14] Sihua Shao, Abdallah Kheirishah, and Juan Paez. 2019. Passiveretro: Enabling completely passive visible light localization for IoT applications. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*. IEEE, 1540–1548.
- [15] Miguel Chávez Tapia, Talia Xu, Zehang Wu, and Marco Zúñiga Zamalloa. 2022. SunBox: Screen-To-camera communication with ambient light. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 6, 2 (2022), 1–26.
- [16] Purui Wang, et al. 2020. Renovating road signs for infrastructure-to-vehicle networking: a visible light backscatter communication and networking approach. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–13.
- [17] Brett Warneke, et al. 2001. Smart dust: Communicating with a cubic-millimeter computer. *Computer* 34, 1 (2001), 44–51.
- [18] Yue Wu, et al. 2020. Turboboosting Visible Light Backscatter Communication. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 186–197.
- [19] Talia Xu, et al. 2022. Exploiting Digital {Micro-Mirror} Devices for Ambient Light Communication. In *NSDI*. 387–400.
- [20] Xieyang Xu, et al. 2017. PassiveVLC: Enabling Practical Visible Light Backscatter Communication for Battery-free IoT Applications. In *MobiCom*. 180–192.
- [21] Zhice Yang, et al. 2015. Wearables Can Afford: Light-weight Indoor Positioning with Visible Light. In *Proceedings of the 13th Annual International Conference on Mobile Systems, Applications, and Services* (Florence, Italy) (*MobiSys '15*). ACM, New York, NY, USA, 317–330.