PolarTag: Invisible Data with Light Polarization

Zhao Tian, Charles J. Carver, Qijia Shao, Monika Roznere, Alberto Quattrini Li, Xia Zhou

Department of Computer Science, Dartmouth College

ABSTRACT

Visual tags (e.g., barcodes, QR codes) are ubiquitous in modern day life, though they rely on obtrusive geometric patterns to encode data, degrading the overall user experience. We propose a new paradigm of passive visual tags which utilizes light polarization to imperceptibly encode data using cheap, widely-available components. The tag and its data can be extracted from background scenery using off-the-shelf cameras with inexpensive LCD shutters attached atop camera lenses. We examine the feasibility of this design with real-world experiments. Initial results show zero bit errors at distances up to 3.0 m, an angular-detection range of 110°, and robustness to manifold ambient light and occlusion scenarios.

CCS CONCEPTS

• **Human-centered computing** → **Ubiquitous and mobile computing**; *Ubiquitous and mobile computing systems and tools.*

KEYWORDS

Visual tags; light polarization

1 INTRODUCTION

Visual tags (e.g., barcodes, QR codes [6, 7]) have become a prominent method of conveying digital information. Existing tags commonly rely on obtrusive visual patterns and are therefore limited in their potential. For example, visual tags are often not embedded into ubiquitous objects, e.g., walls and tables, given their harsh, geometric patterns which are essential for encoding data. Some methods of imperceptibly encoding data include using screens [15, 17] or UV ink [9], complicating the tags with active components or requiring special manufacturing processes.

We propose *PolarTag*, a new design paradigm of passive visual tags which utilizes light polarization to unobtrusively encode data. PolarTag consists of a thin, inexpensive polarizing film with layers of birefringent materials (e.g., everyday transparent tapes) judiciously applied atop. The birefringent layers disperse the polarization direction of outgoing light rays, resulting in color patterns
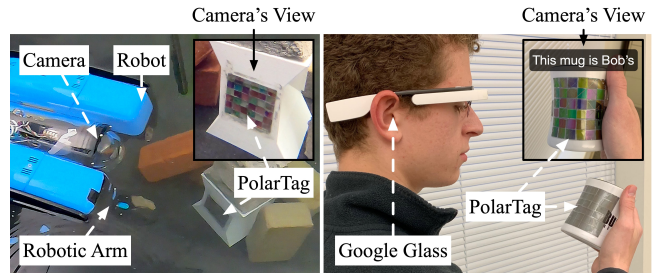

Figure 1: PolarTag in two envisioned applications, underwater robotic construction and augmented reality.

only visible when viewed through a second polarizer (Fig. 1). [1] Such patterns are utilized to encode data. More importantly, the color pattern changes depending on the orientation of the second polarizer. This color-changing feature can be used to boost the robustness of tag detection and support arbitrary tag shapes and designs that are not bound by geometric restrictions.

In this paper, we address the following design challenges: (1) *Encoding*. We implement an efficient encoding scheme that is simultaneously robust against partial occlusion. (2) *Detection*. We propose a method to robustly extract our tag from background scenery at meter distances, exploiting the tag's color-changing nature across different orientations. (3) *Decoding*. We design a decoding scheme that is robust against practical factors, such as ambient light and off-axis viewing. We examine the feasibility of our design with a PolarTag prototype built with low-cost, off-the-shelf components. We summarize our preliminary findings as follows:

- A 12.8 cm × 12.8 cm sized PolarTag with 24 cells encodes up to 40 bits of data and supports decoding distances up to 3.0 m with zero bit errors.
- PolarTag maintains zero-bit decoding errors under various ambient light conditions (7 – 800 lux) and viewing angles between −60° and 50°.
- PolarTag outperforms QR codes in detection robustness with a comparable data-encoding capacity.

Although light polarization has been examined in the past for building an active communication link [5] and facilitating indoor localization [8, 11, 13], our work is the first to explore its use for building passive visual tags that imperceptibly encode data. Unlike RFID tags that require costly RFID readers for decoding data, PolarTag works with cameras widely available on existing mobile devices. As a novel alternative to existing visual tags, PolarTag holds the potential to facilitate various mobile and robotic applications (discussed more in detail in §5) in particularly challenging environments where tags need to be robustly detected and decoded from long distances, wide viewing angles, and low-light conditions.

---

[1]The current PolarTag prototype appears darker than background scenery because of its use of low-cost ($0.80) linear polarizers [1] (42% transmission ratio). A switch to active [10] polarizers with higher transmission ratios (e.g., 90% [2]) can make the tag more transparent and invisible.
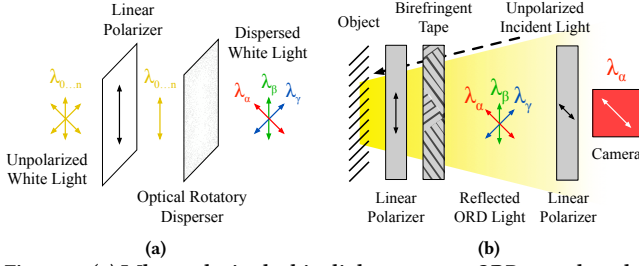
Figure 2: (a) When polarized white light passes an ORD, wavelengths are dispersed into different polarization directions. (b) By placing a polarizer covered with transparent tape layers on objects, reflected light is dispersed into polarization directions and separated by a camera with a second polarizer.
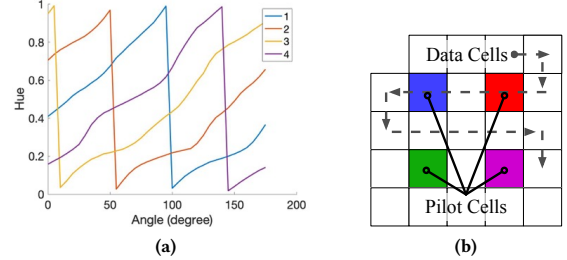

Figure 3: (a) Our chosen tape configurations, separable in the resulting hue in any orientation. (b) The placement of pilot and data cells. The top-left cell is cut out to orient the tag.

2 BACKGROUND AND RATIONALE

Our approach to imperceptibly encode data leverages light polarization patterns that are only discernible when viewed through a polarizer. In particular, we utilize the principle of optical rotatory dispersion (ORD). As shown in Fig. 2a, when polarized white light passes through anisotropic/birefringent materials, such as everyday transparent tape, components of each wavelength composing the white light are dispersed to different polarization directions. This is similar to a prism, which splits white light by wavelength into different spatial directions – in the case of ORD, the white light is split by wavelength into polarization directions. When the dispersed white light is viewed via a second linear polarizer (referred to as an *analyzer*), light appears a single color with the polarization direction that aligns with that of the analyzer. As the analyzer is rotated, the wavelength passing through the analyzer changes, causing the observed color to change as well.

As shown in earlier work [11], the amount of dispersion in polarization directions for each wavelength depends on the thickness and orientation of the birefringent material. Thus, a polarization pattern can be formed by manipulating the number of layers and orientation of the transparent tape across different cells of a polarizer film. When the polarizer film with prearranged transparent tape layers is placed atop an everyday object (Fig. 2b), the ambient light reflected by the object's surface passes through the polarizer and tape layers, exhibiting a color pattern only visible when viewed through another polarizer. Such patterns are the basis for building passive tags that can encode data.

3 POLARTAG DESIGN

When utilizing polarization patterns for designing PolarTag, the main challenge arises from the color-changing nature of the patterns across orientations. Additionally, the tag design should achieve encoding efficiency and decoding robustness. We next present our initial design to address these challenges.

Encoding. We encode data by dividing the tag into cells, where each cell is assigned a tape configuration (e.g., a combination of tape thickness and orientation that leads to a dispersion pattern of light polarization) that encodes a few bits. With M tape configurations, each cell encodes $\log_2 M$ bits. A tag with N cells can then encode $N \log_2 M$ bits in total. The M tape configurations are carefully chosen so that they never result in the same color under

any given orientation. However, since a tag's color pattern changes depending on its relative orientation to the camera, we would need a lookup table that stores the color of each tape configuration in all possible 3D orientations. Decoding the tag then requires knowing tag's orientation to search the lookup table, identify the tape configuration of each cell, and recover the correct bit stream – a process which is computationally expensive and requires heavy training.

To address this challenge, we propose placing a reference cell for each tape configuration, referred to as a *pilot cell*, in a known position on the tag. Remaining cells are *data cells* that actually encode data. Simply comparing each data cell to the pilot cells can identify the tape configuration of each data cell and recover its encoded bits. Although this design slightly lowers the encoding capacity to $(N - M) \log_2 M$ bits per tag, it greatly reduces the decoding overhead and removes the need for expensive training across environments.

Given this design choice, two dominant questions arise. (1) *How many tape configurations (i.e., M) should we have?* The configuration of M faces a tradeoff – a larger M allows more bits encoded per data cell but shrinks the color difference among tape configurations and results in less reliable color detection. Our implementation sets $M = 4$ as this strikes the best balance based on our experiment. Fig. 3a shows the hue values of these four tape configurations under different orientations. (2) *How should pilot cells be arranged relative to data cells?* Since pilot cells must be visible for decoding data cells, we want as many of them to remain unobstructed even if the tag is partially occluded. Clustering M pilot cells together increases the likelihood of them being all blocked under occlusion. Spreading pilot cells across the tag corner's, on the other hand, increases the average spatial distance between pilot and data cells, leading to a greater color deviation between them. Thus, we aim to maximize the minimum distance between pilot cells while excluding the corners. As an example, Fig. 3b shows the optimal placement of four pilot cells for a square-shaped tag with a grid pattern.

To further improve encoding robustness, we add redundancy to our encoding by applying Reed-Solomon (RS) coding [18]. This ensures that data can be decoded even if some data cells are occluded or decoded incorrectly. RS codes are block-based error correcting codes and process each block to correct errors and recover the original data. An RS code configuration $RS(n, k)$ means adding $(n - k)$ parity symbols to every k data symbols to compose a symbol
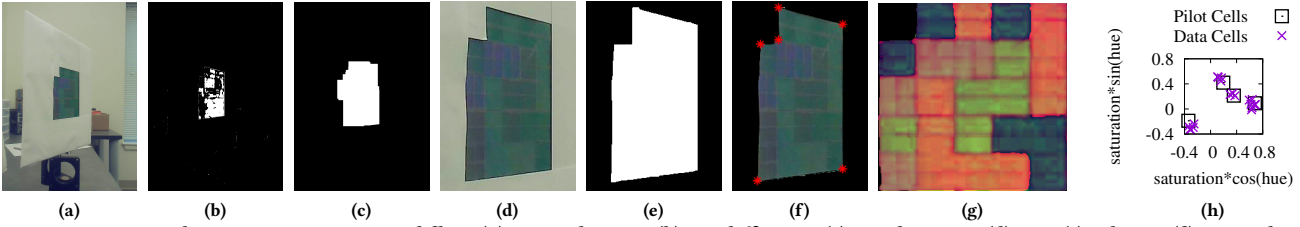
Figure 4: Our complete image processing workflow: (a) original image, (b) H-S difference, (c) ROI detection, (d) ROI, (e) polygon, (f) corner detection, (g) HSV color space, (h) hue-saturation plane, extracting the tag from the background and decoding data.

codeword (n symbols). An RS decoder can correct all combinations of v symbol errors and e symbol erasures[2] provided $2v + e \leq n - k$.

Detection. Tag detection extracts PolarTags from camera images. Existing visual tags solely rely on specific geometric patterns which are vulnerable to interference from background objects with similar shapes or colors. To enhance detection robustness, we exploit the color-changing nature of the polarization pattern – the tag drastically changes its color when viewed at orthogonal polarization directions. Consider two images captured with the second polarizer under two orthogonal polarization directions. Given that ambient light is unpolarized, the color of background objects stays predominantly constant whereas only the tag changes color. Taking the difference of these two image frames allows us to cancel out the background, leaving only the tag.[3] Instead of mechanically rotating the second polarizer at the camera, we propose adding an LCD shutter in front of the camera and supplying a voltage to change the polarization direction.

Fig. 4 shows the complete image processing workflow. We first take two images under two orthogonal polarization directions configured by the LCD shutter. We then evaluate the difference of these two images at the pixel level: we compute both the hue and saturation for each pixel and find the Euclidean distance in the hue-saturation plane. By comparing this distance to a large threshold, we detect the rough outline of a region-of-interest (ROI). Next, we isolate the ROI and take the hue-saturation difference again, this time with a smaller threshold so the tag shape remains intact. To determine these thresholds, we can remove the tag from the environment and take a series of images. For each pair of images (at orthogonal polarization directions), we can determine the maximum and average Euclidean distance in the hue-saturation plane. Since the hue-saturation distance of the tag is expected to be much larger than the environment, we can use these values as a baseline for our thresholds. We then fit the boundary of the ROI to a polygon and use a contour finding algorithm [20] and the Douglas-Peucker algorithm [21] to simplify the number of lines. By computing the perspective transformation (i.e., homography) using the six detected corners,[4] we can transform the tag to its canonical view and extract the colors of each cell for data decoding.

Decoding. The output of the detection phase is used to compare the color of each data cell to the colors extracted from pilot cells. To measure the similarity between colors, we define the distance between two colors as the Euclidean distance in the hue-saturation plane, i.e., $x = \text{saturation} \times \cos(\text{hue})$ and $y = \text{saturation} \times \sin(\text{hue})$. As shown in Fig. 4h, each data cell (x_i, y_i) is assigned to the closest pilot cell $(x_{p^\star}, y_{p^\star})$ using the following equation:

$$p^\star = \operatorname*{argmin}_{p \in \{1, \ldots, M\}} ((x_i - x_p)^2 + (y_i - y_p)^2).$$

Finally, we leverage RS-coding to correct any errors. We decode the tag in the order shown by the arrow in Fig. 3b. Specifically, we start in the upper-left hand corner and read left-to-right. Once we reach the last cell, we move down one row and read right-to-left until all cells are decoded.

4 PRELIMINARY RESULTS

We describe our PolarTag prototype and preliminary evaluation of its performance in a variety of real-world scenarios.

Experimental Setup. We fabricate a 12.8 cm × 12.8 cm prototype using off-the-shelf tape (Home 360 1/2" × 1000" Transparent Tape, $2) and a linear polarizer (3DLens P200, <$1). We select four tape configurations and thus encode two bits per data cell. Before laying the tape strips, we cover the polarizer with a layer of tape at 45° and another at 135°. Given the width of the tape, each square cell is 2.5 cm wide. We arrange the cells in a 5 × 5 grid, removing the top-left cell to detect the tag's canonical orientation. We choose the configuration $RS(15, 10)$ with 4-bit symbols to encode data, resulting in 5 data symbols and 5 parity symbols.

We place our tag flat on a 12" × 12" rigid board, masked with white paper to remove any obvious border contrast. We capture tag images using a USBFHD06H camera ($55) with an Adafruit LCD light valve[5] ($3) placed in front of the camera lens. The shutter and camera are connected to a Raspberry Pi 3 ($28) that saves raw images onboard. Image processing is performed offline using libraries in MATLAB and Python and we discuss real-time processing in §6. We use the C++ implementation of the Schifra RS code library [3] to decode the tag. For each experiment we collect 20 images, 10 at both polarization directions.

Encoding Capacity. To determine the maximum encoding capacity of our tag, we first measure the smallest decodable cell size. The relationship between the cell size in an image (measured in pixels) and the cell's physical size (measured in meters) can be expressed as $P = (S * f)/d$ where P is the pixel size, S is the physical size, and f is the camera's focal length. To determine the smallest decodable

---

[2]Erasures mean we cannot decode the bits from the color. Possible reasons for erasure include occlusion and color aliasing.

[3]If a cell is occluded, the cell's color stays constant. We mark it as an erasure.

[4]With the top-left cell cut, the tag has six corners in its contour.

[5]First, we remove the manufacturer's polarizers attached to both sides of the LCD light valve. Then, we place a 3DLens P200 polarizer on one side of the valve so that our filters are consistent.
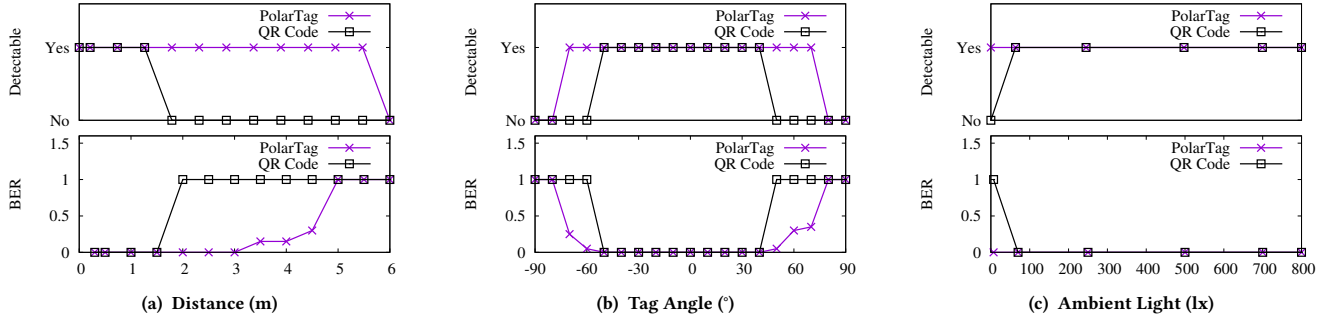
Figure 5: PolarTag and QR codes are compared with respect to decoding robustness (top row) and accuracy (bottom row), across different real-world scenarios including (left) distance, (center) tag orientation, and (right) ambient light conditions.

cell size, we fix S, vary the distance from 0.30 m to 6.00 m, measure f, and compute the bit error rate (BER). As shown in Fig. 5a, we maintain a zero BER up to 3.0 m. At these distances, the pixel size P is 7 px. Although the tag is decodable at longer distances with non-zero BERs, we consider 7 px to be the smallest that we can accurately decode. Since f depends only on the camera settings, we can compute the zero-BER decoding distance for tags with variable cell sizes. We fix the tag size, vary the cell size from 0.2 cm to 3 cm, and compute the maximum zero-BER decoding distance. As shown in Tab. 1, smaller cells pack more bits in a tag but require shorter distances to decode. Overall, the encoding capacity ranges from 26 bits to 8.2 kilobits,[6] with a portion of it used for the actual data (depending on the chosen RS coding configuration).

Table 1: Encoding capacity of PolarTag prototype

Cell size (cm)	0.2	0.3	0.5	1.0	1.5	2.0	2.5	3.0
Distance (m)	0.15	0.3	0.5	1.0	1.5	2.0	3.0	3.5
Number of bits	8182	3630	1300	316	134	70	40	26

Detection Robustness. We next evaluate the robustness of detecting PolarTag, i.e., whether all tag corners can be detected. We use QR codes for baseline comparison by printing a version-2 QR code with medium error correction level (RS coding, 224 out of 625 bits are data bits) in the same size.

We first examine the impact of distance by varying the distance between the camera and the tag from 0.30 m to 6.00 m in 0.50 m increments. As shown in Fig. 5a, PolarTag achieves up to a 5.5 m detection distance, more than three times longer than that of the QR code (1.5 m). We next investigate the angular range of the two tags by fixing them to a ±90° Thorlabs rotational platform (LC1A) and rotating them from −90° to 90°. We fix the camera 60 cm away and align the center of each tag to the center of the camera. As shown in Fig. 5b, PolarTag can be detected from −80° to 80° whereas QR codes are limited between −50° and 40°. Third, we examine different light conditions. We place our tag 42 cm from the camera and turn off all ambient light sources. We then fix a 5500T03 OSRAM LED as the only ambient light source next to the camera and point it towards the tags. We vary the LED's voltage to control the illuminance, measured by a LX1330B light meter. Our camera is set to automatic exposure to detect the tags at low light levels. As depicted in Fig. 5c,

two tags were detectable in the measured range, except for QR codes at 7 lux. Finally, we consider the impact of different background colors in detecting the tag. As shown in Fig. 4a, we used white paper to mask the border of the tag and remove any obvious border contrast that might influence the detection of our tag. We also tested other backgrounds, including multiple colors at once, and found that it has no impact on the detection robustness. This is expected since the tag's background, regardless of its color, is reflecting back unpolarized light which does not skew the tag detection algorithm depicted in Fig. 4. In summary, above results validate the efficacy of adding PolarTag's color-changing feature to enhance the robustness of tag detection.

Decoding Accuracy. We then compare the BER of decoding PolarTag and the QR code under various distances, viewing angles, ambient light conditions (same as for detection), and partial occlusion. The printed QR code is a version 2 code with medium error correction level. We vary the first three conditions in the same manner as that of the detection robustness experiments and Fig. 5 (bottom row) plots the results. We observe that QR codes achieve zero bit errors if they can be detected. On the other hand, PolarTag maintains a zero BER at longer distances (up to 3.0 m), a sufficiently small BER when its orientation is between −60° and 50°, and BER below 0.025 for ambient light levels between 7 lux and 800 lux. In the angular experiment, the BER increases to 0.05 at −60° due to reflections in the image causing a discrepancy in the measured hue. A similar effect also occurs in the ambient light experiment, where reflections due to the LED cause slight decoding errors given the hue ambiguity. However, since the hue of each cell is invariant to the change in brightness, our decoding process succeeds without issues.

Finally, we examine the impact of occlusion on decoding accuracy. We compute the BER for all 11 potential occlusion scenarios by producing (1) symbol errors by changing the hue of specific symbols and (2) symbol erasures by completely masking specific symbols. We utilize the processed image captured at 1.0 m during our distance evaluation and manually change the hue/mask symbols. To examine the efficacy of RS coding, we also consider a baseline scenario without RS coding, where all data cells encode data without adding any redundancy symbols. From the results, we observe a max/min BER of 0.275/0.075 without RS coding. With RS coding, we can decode all scenarios with a zero BER.

---
[6]This is comparable to QR codes, which typically store up to 3 Kb data.

5 SUPPORTING APPLICATIONS

We describe the potential for PolarTag to be utilized in numerous application scenarios.

Augmented Reality. Annotation labels are commonly used in AR applications to add textual or pictorial descriptions to real-world objects. For instance, they can help guide students on a college campus by directly labeling buildings, give additional information about curated objects in a museum, and help customers better select items in a supermarket [14]. However, the view can become overcrowded with AR labels when many objects are present in the scene at the same time, worsening the user experience. Using PolarTag for embedding information into the display of various objects, as shown in Fig. 1, allows information to be displayed only at specific orientations (e.g., when objects are in front of users), given PolarTag's orientation-sensitive color property.

Robotic Construction. Robotic systems can support humans in masonry constructions, where complex structures, such as buildings, are built from individual units (e.g., bricks). Typically, ARTags [16] are used to make the detection and pose estimation of these units easier for robot grasping and placing. PolarTag offers numerous advantages over ARTags for robotic construction. (1) Its invisibility to the naked eye will not affect the appearance of the constructed structure. (2) PolarTag encodes more bits and still supports comparable distances; other visual tags are typically limited to 36 bits with 26 bits used for redundancy. (3) The color-changing nature of PolarTag can be used to improve the tag's pose estimation. (4) PolarTag patterns are caused by the inherent material property, making the tags more durable than printed patterns in harsh environments, e.g., underwater. Fig. 1 is an example of PolarTag used in underwater construction.

Security. Using QR codes as a payment method is gaining more and more popularity, especially in China, since they are cheap and easy to operate. However, QR codes present security risks: as shown in our experiments, the same information can be decoded at multiple orientations, ambient light conditions, and distances. This gives attackers flexibility in compromising the system. PolarTag owners, however, can exploit PolarTag's orientation-dependent change in color to limit the decoding to a specific viewing angle by either modifying the tag configuration or the color mapping scheme. Our experiment shows that the hue of a given tag configuration can change by up to 0.1 within an 8° range, sufficiently thwarting attackers outside this small angular range. Furthermore, tag size can be shrunk to reduce the attack distance so that the tag is decodable only if the user is sufficiently close.

6 DISCUSSION

Although this preliminary work shows PolarTag's potential, challenges remain for a robust, scalable implementation.

Shape and Design. Unlike other visual tags, PolarTag is not restricted to be a square for tag detection, as we leverage the change of color caused by orthogonal polarization directions for detection. This allows adaptation to the shape of the underlying object with arbitrary curvature. The only restriction for the underlying object is that the color/pattern is uniform across cells so that the hue of
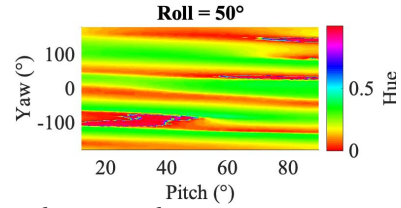

Figure 6: Example regions where our 3D mapping model is insufficient for pose estimation.

pilot cells can be matched to the hue of data cells. Being invisible, a PolarTag could cover an object's entire surface, e.g., a coffee table or wall. This would not only increase tag's encoding capacity but also improve decoding robustness given the larger tag size. New encoding schemes are needed for pilot cells to be always detectable, regardless of orientation or skew of the non-rectangular shape. To locate pilot cells of non-planar tags, pilot cell information can be encoded into strips between adjacent cells. Tag perimeter could also utilize alternating, binary polarization patterns to improve detection.

Automated Manufacturing. Currently, the manufacturing process of PolarTag is not scalable – creating a 5×5 tag takes three hours with manual placement of tape strips atop a polarizer to create each cell. This limits more extensive evaluations (e.g., creation of different tag configurations). We will investigate automated manufacturing methods, including a laser cutter to etch cells of different thickness on a single plastic sheet and a robotic arm for laying strips.

Pose Estimation. We will exploit the orientation-color-changing feature of PolarTag – currently used only for enhancing detection robustness – for improving existing pose estimation. One method is based on empirical measurements to find the mapping between colors and the 3D tag orientation. This presents two main limitations: (1) Preliminary experiments show that multiple regions exist where the hue is invariant to 3D orientation. As shown in Fig. 6, we plot the yaw and pitch for a given roll (50°) and can see multiple regions where the hue is invariant, e.g., when yaw is between 75° and 125° and pitch between 20° and 60°. Consequently, it would be impossible to determine the 3D orientation from the hue alone. (2) Covering the full space would take too long. Assuming it takes 2 s to rotate the tag to a specific angle with a robotic arm and take a picture, it would take 135 days to generate a 3D model within the range of −90° to 90° on all three axes. Another method would be to employ a model that takes into account, for example, the thickness of the birefringent material, the range of incident light angles per cell, and environmental factors such as illuminance. Note that, unlike the 2D model proposed by [8], the model should support arbitrary yaw, pitch, and roll orientations within the 3D space. Finally, instead of relying solely on hue, the model should support saturation and brightness to reduce the number of regions where the mapping is insufficient.

Real-Time Decoding. Although our current algorithms are implemented offline, the computation time is small enough to be implemented in real time. Specifically, if we capture a video at 30 FPS, our LCD shutter is fast enough to switch the polarization state between adjacent frames, meaning the effective detection rate is

15 FPS. Furthermore, our image processing needed to extract the tag typically takes between 0.4 s and 0.8 s and the decoding takes 0.1 s, both implemented in MATLAB. The total time to capture and decode, therefore, is less than 1 s. As a proof of concept, we built a real-time demo of the PolarTag system using a Raspberry Pi, USB camera, LCD shutter, and host computer [12]. The Raspberry Pi streams video frames at orthogonal polarization directions to the host computer over a TCP connection. The host computer then processes the frames in Python and displays the decoded data with minimal lag. Implementing the same functionality in a more efficient language, such as C, will lower the computation time and improve the real-time functionality of the PolarTag system.

Impact on Camera Functionality. To use PolarTag in the real world, mobile devices need to be equipped with LCD shutters in front of their cameras (which, for the first round of applications, can be simply a mobile phone case with built-in hardware). Currently, these shutters are fast enough to support 30 or 60 FPS, meaning video recording would be unaffected. The brightness of captured images would be slightly lower than without the shutter, since a portion of unpolarized light would be blocked by the filter. To address this issue, image processing methods could be implemented or, in the case of multi-camera phones (e.g., the iPhone 11), the shutter can be attached to a single camera while the others are used for normal video capture. Finally, the power consumption of these shutters is negligible compared to other components on the mobile device. In our implementation, we alternate the shutter voltage between 0 V and 3.3 V (drawing $\approx 10\mu A$).

7 RELATED WORK

Polarization. Prior works studied the generation of a passive light polarization pattern at the luminary to facilitate indoor localization [8] and augment inertial tracking [11]. Specifically, Li et al. [8] derived a model to describe relationship between light polarization pattern and the direction. Tian et al. [11] used the polarization pattern to augment indoor inertial tracking. Unlike our work, these designs have not examined the use of polarization patterns for data encoding/decoding. Wang et al. [13] and Chan et al. [5] studied polarization-based modulation schemes for indoor positioning or light communication. These works build an active unit to communicate dynamic data, while our work designs passive visual tags that encode static data and can be attached to everyday objects.

Visual Tags. Matrix barcode such as QR codes[6, 7] are popular due to their fast readability and good storage capacity. Fiducial marker systems, which enjoy advantages in uniqueness and flexibility, are widely used in augmented reality (AR), robotics and simultaneous localization and mapping (SLAM). So far, representative marker techniques include ARTag [16], AprilTag [19], and CalTag [4], providing a promising way for object detection and tracking. However, their obtrusive geometric patterns decrease the overall user experience and raise the risk of information leakage. Some prior works have reduced tag obtrusiveness with digital devices or complex manufacturing techniques. Specifically, Kennedy et al. [17] utilized digital screens to imperceptibly encode information. Furthermore, Meruga et al. [9] encoded information with

UV ink that was only visible under certain viewing conditions. Compared to these methodologies, our tag design is easier to manufacture, is completely passive, and does not require energy to unobtrusively encode data.

8 CONCLUSION

We presented PolarTag: a novel, invisible, passive tag based on light polarization. The prototype used low-cost, off-the-shelf components and camera, resulting in an overall cost under $100 (<$1 for the tag). Initial results showed PolarTag achieving zero bit errors up to 3.0 m and an angular range of 110°, outperforming QR codes and making it a promising unobtrusive marker for mobile and robotic applications.

9 ACKNOWLEDGEMENTS

REFERENCES

[1] [n.d.]. 3DLens Polarizers. https://3dlens.com/linear-polarizer-film.php. Accessed: 2019-10-17.
[2] [n.d.]. Polarization conversion system. https://www.imagineoptix.com/technology/polarization-conversion-system/. Accessed: 2019-10-22.
[3] [n.d.]. Schifra Reed-Solomon error correcting code library. http://www.schifra.com. Accessed: 2019-10-16.
[4] Atcheson et. al. 2010. CALTag: High Precision Fiducial Markers for Camera Calibration. In *Vision, Modeling, and Visualization*.
[5] Chan et al. 2017. Poli: Long-range visible light communications using polarized light intensity modulation. In *Proc. of MobiSys*.
[6] Kan et al. 2009. Applying QR code in augmented reality applications. In *Proc. of the International Conference on Virtual Reality Continuum and its Applications in Industry*.
[7] Kieseberg et al. 2010. QR code security. In *Proc. of the International Conference on Advances in Mobile Computing and Multimedia*.
[8] Li et al. 2018. RainbowLight: Design and Implementation of a Low Cost Ambient Light Positioning System. In *Proc. of MobiCom*.
[9] Meruga et al. 2012. Security printing of covert quick response codes using upconverting nanoparticle inks. *Nanotechnology* 23, 39 (2012).
[10] Seo et al. 2011. 39.2: Polarization Conversion System Using a Polymer Polarization Grating. In *SID Symposium Digest of Technical Papers*, Vol. 42. 540–543.
[11] Tian et al. 2018. Augmenting Indoor Inertial Tracking with Polarized Light. In *Proc. of MobiSys*.
[12] Tian et al. 2020. Demo: PolarTag – Invisible Data with Light Polarization. In *Proc. of HotMobile*.
[13] Wang et al. 2015. Wearables Can Afford: Light-weight Indoor Positioning with Visible Light. In *Proc. of MobiSys*.
[14] Wang et al. 2017. ARShop: A Cloud-based Augmented Reality System for Shopping. *PVLDB* 10 (2017), 1845–1848.
[15] Yamanari et al. 2009. Electronic Invisible Code Display Unit for Group Work on Reminiscence Therapy. In *Proc. of the International MultiConference of Engineers and Computer Scientists*.
[16] Mark Fiala. 2005. ARTag, a fiducial marker system using digital techniques. In *Proc. of CVPR*.
[17] John T Kennedy. 2013. Visually imperceptible matrix codes utilizing interlacing. US Patent 8,553,146.
[18] Shu Lin and Daniel J Costello. 2001. *Error control coding*. Pearson Education India.
[19] Edwin Olson. 2011. AprilTag: A robust and flexible visual fiducial system. In *Proc. of ICRA*.
[20] Satoshi Suzuki and Keiichii Abe. 1985. Topological structural analysis of digitized binary images by border following. *Comput. Vision Graph* 30, 1 (1985), 32–46.
[21] Mahes Visvalingam and J Duncan Whyatt. 1990. The Douglas-Peucker Algorithm for Line Simplification: Re-evaluation through Visualization. In *Comput. Graph. Forum*, Vol. 9. 213–225.