Uber-in-Light: Unobtrusive Visible Light Communication Leveraging Complementary Color Channel

Mostafa Izz*, Zhongyuan Li†, Hongbo Liu*, Yingying Chen†, Feng Li*

*Purdue School of Engineering and Technology, IUPUI
Indianapolis, IN, 46202
{*mosmoham, hl45, fengli*}*@iupui.edu*

†Dept. of ECE, Stevens Institute of Technology
Castle Point on Hudson, Hoboken, NJ 07030
{*zli37, yingying.chen*}*@stevens.edu*

Abstract—Recently, Visible Light Communication (VLC) over a screen-camera channel has drawn considerable attention to unobtrusive design. It overcomes the distractive nature of traditional coded image approaches (e.g., barcodes). Previous unobtrusive methods fall into two categories: 1) utilizing alpha channel, a well known concept in computer graphics, to encode bits into the pixel translucency change with off-the-shelf smart devices; and 2) leveraging the spatial-temporal flicker-fusion property of human vision system with the fast frame rate of modern displays. However, these approaches heavily rely on high-end devices to achieve both unobtrusive and high accuracy screen-camera-based data communication without affecting video-viewing experience. Unlike previous approaches, we propose Uber-in-light, a novel unobtrusive and accurate VLC system, that enables real-time screen-camera communication, applicable to any screen and camera. The proposed system encodes the data as complementary intensity changes over Red, Green, and Blue (RGB) color channels that could be successfully decoded by camera while leaving the human visual perception unaffected. We design a MFSK modulation scheme with dedicated frame synchronization signal embedded in an orthogonal color channel to achieve high throughput. Furthermore, together with the complementary color intensity, an enhanced MUSIC-based demodulation scheme is developed to ensure highly accurate data transmission. Our user experience experiments confirmed the effectiveness of delivering unobtrusive data across different types of video content and resolutions. Extensive real-time performance evaluations are conducted using our prototype implementation to demonstrate the efficiency and reliability of the proposed system under diverse wireless environments.

I. INTRODUCTION

The abundance of screen and camera-equipped devices makes the visible channel readily available for the device-to-device communication [1], [2]. Quick Response (QR) Code [3], which embeds information into 2D bar-code optical label, is the most popular method for data collection over the visible channel. Active efforts have been devoted to develop innovative bar-codes for improving the data rate atop original media content [4]. However, due to both limited screen size and camera resolution, solutions based on QR code fall into a dilemma of sacrificing the information-carrying capacity by reducing code size for better viewing comfort of audiences.

The limitations of QR code inspire the unobtrusive communication design over visible channel, such as unobtrusive barcodes [5], or integrating images/watermark into barcodes [4]. However, the high complexity on generating the machine-readable optical labels (i.e., barcodes) prevents these approaches from widely adoption for visible communication, especially for large amount of streaming data transmission. In order to overcome the limitations on QR code based visible light communication, HiLight [6] and InFrame++ [7] utilized

specific characteristics of human visual perception system to achieve unobtrusive communication schemes over visible channel. Specifically, HiLight [6] embeds data as small pixel luminance changes over the orthogonal transparency channel (alpha channel) on screen, avoiding affecting human viewing experiences. however, the accuracy and throughput of HiLight are limited by the system design and hardware requirements. InFrame++ [7] multiplexes data onto full-frame video contents through spatial-temporal complementary frames leveraging the temporal flick-fusion property [8] of human vision system. InFrame++ requires high-end monitor with high frame rate (i.e., $120FPS$) acting as the transmitter, which may not be satisfied by many display device.

Given the limitations of existing studies, we develop *Uber-in-light*, aiming to achieve unobtrusive visible light communication with both high accuracy and high throughput over the screen-camera channel. Uber-in-light is a generic solution applicable to any user camera without special hardware requirements just like in the analogy of the Uber car service, in which any customer can obtain a guaranteed car service at any time and location when needed. To achieve the design goals in Uber-in-light, several challenges need to be addressed as follows:

Unobtrusiveness vs. Accuracy: In order to achieve unobtrusive viewing experience for the audience watching the videos, the luminance changes for data representation in our proposed system should be as small as possible to avoid the flicker effects perceived by human vision; on the other hand, small luminance changes would result in low Signal-to-Noise-Ratio (SNR) of the received visible light signals, which largely increases the difficulty on data decoding at camera. As such, it is challenging to solve the above contradiction on luminance manipulation to achieve both unobtrusiveness and high accuracy under various video contents.

Accuracy vs. Throughput: Transmission throughput is one of the critical performance that the communication systems most concern about. Both HiLight and InFrame++ mainly rely on the high resolution of the display screen to achieve high throughput. In particular, the display screen is divided into multiple grids, each acting as an independent visible light channel, to facilitate large amount data transmission in a parallel fashion. But the pursuit of high throughput utilizing large number of grids usually sacrifices the accuracy since the perceived area resolution of each grid on camera decreases as the number of grids increases. Throughput performance is also closely related to the modulation scheme adopted. The modulation rate for HiLight is limited to 1 bit per symbol due

to the adoption of Binary Frequency Shift Keying (BFSK).

Synchronization: Similar to any other types of wireless channels, the visible light channel also suffers from various interfering factors affecting the reliability of the communication systems. Synchronization failure is one of the most serious factors. Even though HiLight and InFrame++ utilize high-end GPU to accelerate the data processing and thereby facilitate video frame synchronization, it still can not completely avoid the frame mismatch between the transmitter and receiver. After all, the screen and camera are not delicately designed for communication purpose, so they must share the resources with many irrelevant applications running on the same platform. Considering the overhead resulted from other irrelevant applications, the processing delay may not be acceptable for video frame synchronization, not to mention the implementation with low-end GPU or CPU.

While HiLight simply encodes the data as the overall luminance changes on screen, our Uber-in-light system enables unobtrusive screen-camera communication through pixel color intensity manipulation. To deal with the above challenges, the following contributions are made in this paper:

- We develop a new unobtrusive screen-camera communication system that encodes the data as complementary intensity changes over different color channels for any screen content. The intensity changes could be successfully interpreted by camera without getting noticed by human vision system. Furthermore, the complementary intensity changes also serve as redundant coding for information reinforcement during data transmission.

- With respect to the screen-camera channel, we propose a Multi-Frequency Shift Keying (MFSK) scheme for modulation and an enhanced Multiple Signal Classification (MUSIC) scheme for demodulation. The schemes are compatible with any hardware platforms and ensure data transmission with high accuracy and throughput.

- We develop a new channel synchronization method leveraging an orthogonal color channel, which periodically transmits synchronized signals, to prevent frame mismatch during the data transmission, thereby boosting the reliability of the proposed system.

- The proposed Uber-in-light system is prototyped using off-the-shelf devices, and validated with extensive experiments in terms of user perception, accuracy and throughput across a variety of environments and platforms.

The rest of the paper is organized as follows. We introduce the related work in the area of visible light communication in Section II. In Section III, we present our proposed Uber-in-light system over screen-camera channel. Next, we describe the real implementation of the proposed schemes in Section IV. The extensive performance evaluation is conducted through real experiments in Section V. Finally, we conclude our work in Section VI.

II. RELATED WORK

Visible light communication (VLC) has attracted increasing attention in recent years because it utilizes the visible light spectrum to carry information. Based on the type of transceiver equipment, existing research studies fall into two major categories: LED-based VLC and screen-camera VLC. Specifically, LED-based VLC modulates the transmitted data as the intensity of the emitted light signals via the LED optical source [9]–[11], and photodiodes at the receiver side captures the data through detecting the emitted light intensity variations. For screen-camera communication, the transmitted data is encoded as visual images displayed on screen, and any camera-equipped device facing the screen could extract the data by recognizing the images.

There are active research in screen-camera communication with visible coded images as the data carrier. Many works are dedicated to improve the reliability and performance of the visible light communication over the screen-camera channel [2], [12], [13] such as Lightsync and Styrofoam. Cobra [14] and Pixnet [15] focus on boosting the data throughput. SBVLC [1] ensures the security of visible light channel by manipulating screen view angles and leveraging user-induced motions. Strata [16] and Bokode [17] propose to realize barcode detection at a relatively large distance. The visible-coded-image-based VLC shows significant advantages on transmission throughput and accuracy, however, the space reserved for displaying the coded images always interferes with the content currently shown on the screen and may create unpleasant viewing experiences to the audience.

In order to overcome the unpleasant viewing experiences during screen-camera communication, unobtrusive screen-camera communication emerges. HiLight [6] encodes data into small pixel translucency changes on top of arbitrary screen contents (e.g., image, video clip, game console, or application GUI, etc.). The accuracy and throughput of HiLight heavily relies on the resolution on both screen and camera. Another recent work InFrame++ [7] multiplexes data onto full-frame video contents through spatial-temporal complementary frames leveraging the temporal flick-fusion of human vision system to achieve unobtrusive communication with high data rate and accuracy, but it can only be applied to the display devices (i.e., screen) supporting high frame rate (i.e., over 120 Hz).

Furthermore, some image processing related techniques, such as steganography and invisible digital watermarking [8], [18], also convey certain information over the original digital media (e.g., images or videos) with specific changes imperceptible to human eyes. But unlike visible light communication, the manipulated digital media with steganography and watermarking serve for data security, authenticity verification or integrity checking [19], which can only be decoded with specific algorithms. Thus, it does not guarantee that they can be captured by the camera for data delivery wirelessly.

Different from the existing studies on unobtrusive visible light communication, the proposed Uber-in-light system utilizes separated color channels with complementary intensity changes to perform data transmission, which could be captured by camera but imperceptible to human vision systems. Built upon the unobtrusive design, our system also develops advanced data modulation and demodulation schemes, which can be readily implemented on any commercial devices, to achieve high data throughput as well as accuracy with respect to screen-camera channel.

III. SYSTEM OVERVIEW

In this section, we first study the color perception on human vision system (HVS) and camera, which serves as the
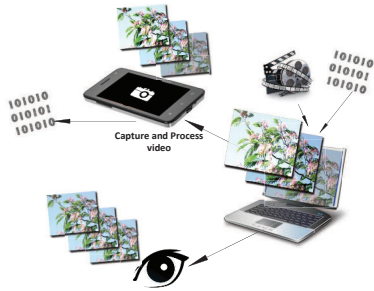
Fig. 1. System Overview

foundation of unobtrusive screen-camera communication, and then introduce the working flow of Uber-in-light.

A. Background and Feasibility Study

Most of current electronic display screens rely on *RGB color model*, which adds three primary colors (i.e., Red, Green and Blue) together with different color intensity combinations, to reproduce a broad range of colors. Particularly, each pixel on the screen consists of three small and very close but still separated RGB light sources. These separate sources are indistinguishable at common viewing distance. As a result, the human vision system will only capture a solid color without differentiating each primary color and the corresponding intensity. Basically, the human vision system interprets the color based on three attributes: saturation, hue and luminance. And the human vision system is more sensitive to luminance changes [20] than other attributes, which is defined as the weighted summation of intensity changes of three primary colors. Therefore, regardless of the luminance variation of original screen content (e.g., video clip), *unobtrusiveness* on human vision system is mainly in pursuit of minimum luminance changes resulted from other luminance sources.

Unlike human vision system, the digital camera system is equipped with CMOS or CCD image detectors, which are formed by an interlaced grid of red, green, and blue pixel sensors in high density, thus, each primary color intensity could be identified independently with selected pixel sensors. Inspired by the differences on color perception between human vision system and camera system, if the intensity changes that are applied to different color channels contribute to the zero luminance changes, the unobtrusiveness over screen-camera channel is achieved. Specifically, in this work we adopt complementary intensity changes among the three primary color channels to keep the overall luminance unaltered. For example, given the original color intensity combination for a particular screen content as $(100, 70, 200)$, the luminance equals to $100 + 70 + 200 = 370$ if the intensity on each color channel equally contributes to the luminance. If the color intensity combination is updated as $(98, 70, 202)$ with small intensity changes $(2, 0, -2)$, the luminance remains.
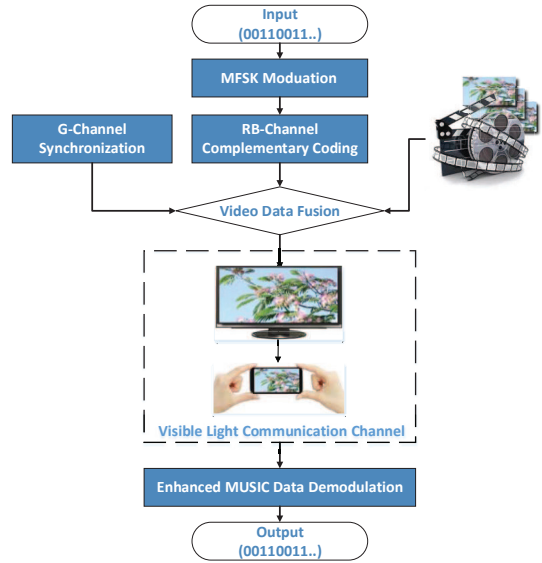

Fig. 2. Illustration of System Flow

Furthermore, the luminance of original screen content usually has little variation during a short period due to the strong internal correlation between adjacent frames. It thus provides the opportunities to extract the encoded intensity changes from consecutive frames by camera.

In order to verify the feasibility of the proposed luminance-based approach, 12 people are invited to watch the videos with and without complementary color intensity changes and evaluate the difference according to different perception scores defined in Table I. As the results shown in Table II, it is encouraging to find that the averaged perception score from the 12 people would achieve as high as 4.6, which means the users could hardly sense any difference between the original and modified videos. In contrast, if the color channels area updated with independent intensity changes, it is easy to be detected by the human vision system. Note that *Independent* means Red, Green and Blue channels are modified independently with different color intensity changes, *Complementary* means that complementary intensity color changes are applied, and *Original* means the original video.

B. System Flow

Our proposed Uber-in-light seeks to multiplex the bit stream onto a specific pattern of RGB color intensity changes on screen, which could be accurately decoded by camera without affecting the luminance perceived by human vision system. The proposed Uber-in-light as shown in 1 is made up of the transmitter (i.e., screen) and receiver (i.e., camera) connected with each other through the visible light channel. There are four major components in the system as shown in Figure 2: Multi-Frequency Shift Keying modulation, RB-channel complementary coding, G-channel synchronization and Enhanced MUSIC-based demodulation.

Given a bit stream and pre-defined grids on the screen, the transmitter first modulates these bits using Multi-Frequency

TABLE I
PERCEPTION SCORE DEFINITIONS

Perception Score	Description
1	Strong flicker or noticeable artifacts
2	Less strong flicker or noticeable artifacts
3	Merely noticeable artifacts
4	Very few noticeable artifacts
5	Certainly no noticeable artifacts

TABLE II
FEASIBILITY STUDY ON UNOBTRUSIVENESS

Setting	Independent	Complementary	Original
Perception Score	2.2	4.6	5

Shift Keying (MFSK). Particularly, the bit stream is partitioned into multiple bit segments of a certain length (e.g., $010111\cdots \Rightarrow 010, 111, \cdots$), and then each segment (e.g., 010 or 111) will be mapped to a unique frequency (e.g., $10Hz$ or $12Hz$), which is utilized to generate the modulated signal in the form of periodic color intensity changes. Compared with BFSK in HiLight, MFSK modulation enables each frequency to represent more bits during data transmission, which largely increases the visible light channel capacity. However, simple MFSK modulation does not ensure unobtrusive data transmission, so next the color intensity of the modulated signals is adjusted with our *channel complementary coding scheme*. In particular, our channel complementary coding scheme chooses Red and Blue channels for data transmission, while Green channel is utilized for transmitting the synchronization signal. The modulated signals are encoded as complementary color intensity changes over Red and Blue channels to provide the redundancy for accurate decoding (i.e., δ and $-\delta$ for Red and Blue channel respectively). In order to synchronize the data transmitting and receiving, a dedicated signal with periodic color intensity changes on the Green channel is integrated with data transmission. Furthermore, the color intensity changes on the Green channel also complement the overall color intensity changes on both Red and Blue channels to guarantee the unobtrusiveness of data transmission. Before the data is emitted into the visible light channel, the signals on both data and synchronization channels integrate the original screen content to be displayed on the screen.

At the receiver side, the camera captures the video frames carrying the data of interest and extracts the encoded intensity changes on each individual color channel. our *enhanced MUSIC-based demodulation scheme* identifies the frequency information embedded in color intensity changes, and then maps these frequencies to their corresponding bit segments to recover the original bit stream.

IV. System Design

In this section we introduce the design details of Uber-in-light. Our system works as follows: the input bits stream is modulated as a complementary color intensity changes over RGB channels on screen along with dedicated synchronization signal, and then such color intensity changes are demodulated at the receiver side leveraging high resolution frequency estimation algorithm. In our design, the proposed modulation and demodulation algorithms are compatible to any hardware platform of screen and camera.

A. Multi-Frequency Shift Keying Modulation On Color Intensity Changes

Once the input bit stream arrives at the transmitter, the data is modulated onto the carrier signals of a certain frequency, which are represented as periodic color intensity changes over a certain number of frames displayed on the screen. In order to boost the throughput of Uber-in-light, we develop the MFSK modulation scheme over the visible light channel to involve more frequencies for data modulation so that more bit information will be carried at each frequency.

To facilitate the modulation on the incoming bit stream, our system first partitions the bit stream into multiple bit segments of a certain length, and each bit segment will be associated with a particular frequency of modulated signal based on predefined mapping relationship as shown in Table III. Particularly, gray code is adopted to ensure that two successive frequencies are encoded with only one bit difference so that the proposed system will be less susceptible to interference on the visible light channel. According to Nyquist sampling principle, there should be at least two points per cycle period of the highest frequency used for modulation, therefore, the maximum frequency of the carrier signal should be at most half of the scanning rate of a camera. For example, if the scanning rate of the camera is $30Hz$ in our system, the maximum frequency for modulation is $15Hz$. Furthermore, the frequency chosen for modulation can not be too small, otherwise the modulated signal completes fewer cycles during a single bit segment period, which may cause the demodulation process less reliable in recognizing this particular frequency at the receiver. The minimum single bit segment period is $300ms$ in our system, so the modulated signal with $6Hz$ can not even complete two cycle periods during the single bit segment period (i.e., $1/6Hz \times 2 \approx 334ms > 300ms$). Therefore, we choose 8 different carrier frequencies ranging from $7Hz$ to $14Hz$ to perform modulation on different bit segments. To further boost the channel capacity, the screen will be divided into multiple rectangular grids, each grid acting as an independent visible light channel to perform the MFSK data modulation following the above procedures.

B. RB-Channel Complementary Coding

Simply modulating the data into constant color intensity changes cannot guarantee the unobtrusiveness of screen-camera communication. Thus, the modulated signals should be adjusted on the amplitude of the channel color intensity. We develop the RB-channel complementary coding scheme to further process the modulated signals before emitting them to the visible light channel.

The basic idea is to achieve equivalent color intensity changes when applying different signs to Red and Blue channels. Since the red, blue and green channels are orthogonal to each other, they contribute to the overall luminance together. The proposed RB-Channel based complementary coding scheme serves for two main purposes: 1) it maintains overall luminance changes from both Red and Blue color channels with little changes such that the modulated signals cannot be perceived by human vision system, thereby achieving the unobtrusiveness design goal; 2) since the Red and Blue channels are encoded with complementary color intensity changes, it will provide the channel redundancy to ensure the data being accurately decoded at the receiver. To illustrate how the proposed scheme works, we first employ the similar definition of luminance as describe by Reinhard, et.al [21], which is a weighted combination of color intensity on RGB channels shown in Equation 1.

$$\begin{aligned} L &= L_R + L_G + L_B \\ &= 0.2126 \times R + 0.7152 \times G + 0.0722 \times B, \end{aligned} \tag{1}$$

TABLE III
BIT SEGMENT-FREQUENCY MAPPING WITH GRAY CODING

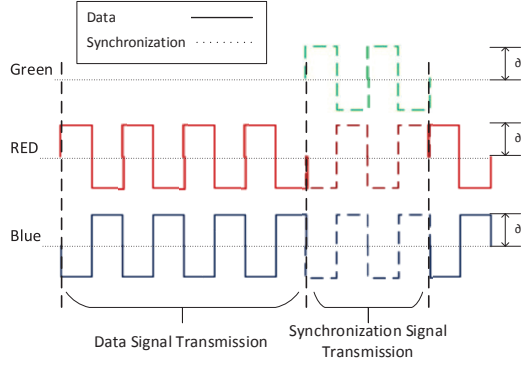| Bit Segment	Frequency (Hz)	Bit Segment	Frequency (Hz)
000	7	001	8
011	9	010	10
110	11	111	12
101	13	100	14

Fig. 3. Illustration of complementary color intensity changes.

where L_R, L_G and L_B are the luminance components from red, green and blue channels, and R, G and B represent the color intensity on the red, green and blue channel respectively [22], and L is the luminance value derived based on the intensity on RGB color channels. From Equation 1, we find that red and blue channels together contribute much less to the luminance than that of the Green channel, thus, small color intensity changes on red and blue channel will not disturb the original screen content much. Therefore, we choose the red and blue channels as the data channel and represent the modulated signals with complementary color intensity changes on the two-color channels as shown in Figure 3. Particularly, the parameter δ is introduced to control the color intensity changes, and the overall luminance is updated as follows:

$$L^* = 0.2126 \times (R - \delta) + 0.7152 \times G + 0.0722 \times (B + \delta),$$
(2)

where L^* represents the updated luminance value after applying the RB-based complementary color intensity changes. The overall luminance changes are $0.1404 * \delta$ compared with the original luminance in Equation 1. We note that a larger δ would destroy the original screen content even if the overall luminance does not change much, whereas a smaller δ would result in low detection accuracy at the receiver. Hence, δ should be carefully designed in real implementation. In our system, we set $\delta = 0.8\%$, which makes the luminance change as low as 0.1% so that the unobtrusiveness will be guaranteed during data transmission.

C. G-Channel Synchronization Scheme

Since the screen and camera are not specifically designed for visible light communication, synchronous data transmission and receiving can not be achieved without external control information. A dedicated color channel must be allocated to synchronize the communication between transceivers. From Equation 2, the channel complementary coding scheme involves both red and blue channels but leaves the orthogonal green channel unaltered. Therefore, we propose to utilize the green channel to transmit synchronization signals simultaneously with the modulated data signals for reliable screen-camera communication.

The synchronization signal on the green channel is also in the form of periodic color intensity changes over a certain number of frames. But unlike the data signals in red and blue channels, the synchronization signal is transmitted intermittently as shown in Figure 3. Without the synchronization signal on the green channel, the luminance almost does not change with the data signals since complementary color intensity changes only on red and blue channels. When the synchronization signal appears on green channel without updating the color intensity changes on other two color channels, the unobtrusiveness will not be maintained based on the luminance model in Equation 2. Therefore, when both data and synchronization signals are transmitted on the screen-camera channel, the color intensity changes on the three color channels should be adjusted again to maintain low luminance changes. We use the same parameter δ to indicate the color intensity change on the green channel, and the luminance Equation 2 is further updated as follows:

$$L^s = 0.2126 \times (R - \delta) + 0.7152 \times (G + \delta) \\ + 0.0722 \times (B - \delta),$$
(3)

where L^s represents the luminance integrating the G-channel synchronization signal. Equation 3 indicates that the color intensity changes on the synchronization channel (i.e., green channel) will be encoded to complement the overall color intensity changes on the data channel (i.e., red and blue channels). For example, given $\delta = 0.4\%$, the final luminance change will be close to 0 (i.e., $0.4304 * \delta = 0.17\%$), which ensures unobtrusive data transmission against arbitrary screen content. Since the synchronization channel (i.e., green) is orthogonal to the data channel (i.e., red and blue), the receiver does not have difficulty on distinguishing the color intensity changes utilized for data transmission and synchronization information. Finally, the transmitter design is completed by displaying the video stream on the screen together with the modulated data signals on red and blue channels and the synchronization signal on the green channel.

D. Enhanced MUSIC-based Demodulation Scheme

After the modulated video frames captured by the receiver (i.e., camera), the color intensity changes on each color channel will be extracted first and passed to the MFSK demodulator to recover the bit stream. Figure 4 summarizes the main steps of data demodulation process in Uber-in-light. The basic idea is as follows: The receiver keeps recoding incoming video frames displayed on screen and store them in a predefined demodulation window. Then the adjacent frames within the demodulation window are examined to derive the frequencies of periodic color intensity changes, which are the modulated signal carrying data and synchronization information, and finally the original bit stream is recovered with the enhanced MUSIC-based demodulator by mapping the extracted frequencies to the corresponding bit segments.

Color Intensity Changes Extraction. The receiver in our system takes three steps to complete the extraction of color intensity changes from the captured video stream: grid identification, color channel separation and temporal correlation-based intensity change detection. *1) Grid Identification:* Since the screen is divided into multiple grids to perform parallel data transmission, in order to receive the multiple data streams simultaneously, we adopt the well-known camera calibration methods [23], which deploys visible chessboard patterns appearing shortly on the screen, to locate and rectify the screen borders, and then each grid on the chessboard is also identified. *2) Color Channel Separation:* The data has already been embedded into different color channels of
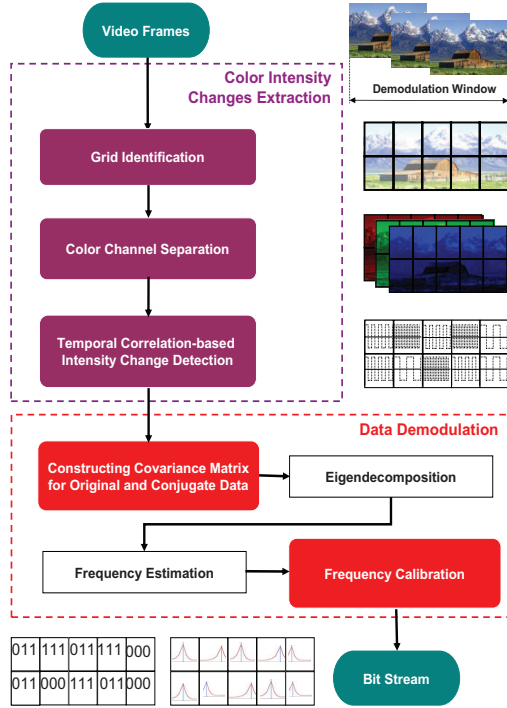
Fig. 4. Illustration of enhanced MUSIC-based demodulation scheme.

the captured video stream. Given any identified grid, the receiver will distinguish each individual color channel with the pixel sensors of different colors to facilitate the subsequent intensity change detection. *3) Temporal Correlation-based Intensity Change Detection:* The color intensity change extraction is established on the temporal-correlation between two adjacent frames $F(k)$ and $F(k+1)$. Without loss of generality, we assume a single grid applied on the two frames. The camera first extracts the color intensity on each pixel of the two captured frames, and then calculates the difference on color intensity across all the pixels between $F(k)$ and $F(k+1)$. For example, if the pixel at the location $(0,0)$ has the color intensity $(R, G, B)_k = (100, 70, 200)$ on $F(k)$ and $(R, G, B)_{k+1} = (98, 69, 202)$ on $F(k+1)$, the color intensity change between these two frames of the same pixel is derived as $(R, G, B)_{k+1} - (R, G, B)_k = (-2, -1, 2)$ for three different color channels respectively. Summing up the color intensity changes across all the pixels renders the overall color intensity changes between two adjacent frames. Since the transmitted data is encoded as a pair of complementary color intensity changes over red and blue channels, we define the extracted color intensity changes as $D = \delta_B - \delta_R$, which almost doubles the color intensity changes compared with that in a single color channel. Therefore, the reliability of data demodulation could be largely reinforced. The color intensity changes in the green channel is also extracted in a similar way to precisely control the frame rate synchronized between the screen and camera.

Data Demodulation. Next, the receiver will perform demodulation by estimating the frequency of extracted color intensity changes across the captured frames. The screen-camera channel suffers from a variety of vulnerabilities preventing accurate data demodulation including variable contents on the screen, ambient lighting interference, and thermal noises

inside the camera, etc. Previous works, such as HiLight, adopt the Fast Fourier Transform (FFT) technique to perform demodulation from the extracted color intensity changes, but they do not show high demodulation accuracy due to the low spectrum resolution of FFT and severe background noise on the screen-camera channel. The MUltiple SIgnal Classification(MUSIC) [24] is a high resolution spectrum analysis algorithm leveraging the orthogonality between signal and noise subspace derived from eigendecomposition of signal covariance matrix. The performance of MUSIC relies on the estimation accuracy on the noise subspace. In order to achieve high accuracy on frequency estimation, we propose an enhanced MUSIC (MUltiple SIgnal Classification) demodulation scheme.

We first define the signal model, which is abstracted from color intensity changes on the transceiver, as $X(t) = S(t) + n(t)$, where $X(t)$ is the received signal varying along time t (i.e.,a vector of the color intensity changes extracted at receiver), $S(t)$ is the transmitted signal (i.e., a vector of color intensity changes encoded at transmitter), and $n(t)$ is the additive background noise. Then the covariance matrix R_x based on the signal model is derived as:

$$R_x = E\left[X(t)X^H(t)\right] \overset{(a)}{=} U_s \Lambda_s U_s^H + U_n \Lambda_n U_n^H, \quad (4)$$

where H represents conjugate transpose operation, (a) is based on the eigendecomposition of the covariance matrix. Since $X(t)$ is single-frequency signal within a demodulation window, Λ_s is the largest eigenvalue of R_x, U_s is the eigenvector corresponding to Λ_s, and U_n is the matrix that contains the eigenvectors associated with the remaining eigenvalues. U_s and U_n are defined as signal and noised subspace respectively. Due to the unobtrusiveness design in the proposed system, $S(t)$ has smaller amplitude in comparison with the background noise $n(t)$, which would result in low signal-to-noise ratio (SNR). To mitigate the impact from low SNR on frequency estimation, we propose to enhance the MUSIC algorithm by extending the original signals with conjugate covariance matrix R_y [25] as follows:

$$R = R_x + R_y = E\left[X(t)X^H(t)\right] + E\left[Y(t)Y^H(t)\right] \\ = V_s \Lambda_s' V_s^H + V_n \Lambda_n' V_n^H, \quad (5)$$

where $Y(t) = JX(t)$, J is anti-diagonal identity matrix, V_s and V_n represent signal and noise subspace of R, and Λ_s' and Λ_n' are the diagonal matrices containing the eigen values corresponding to signal and noise subspace. According to matrix theory, the updated covariance matrix R has the same noise subspace as R_x (i.e., $U_n = V_n$), but provides more accurate estimation on noise subspace [26]. Leveraging the orthogonality between signal subspace and noise subspaces, the frequency of $X(t)$ is obtained through searching for f that maximizes the objective estimator function as follows:

$$\arg \max_f \ \frac{1}{e^H(f)(V_n V_n^H)e(f)}, \quad (6)$$

where $e(f)$ is the steering vector in the form of $[0, e^{j2\pi f/\phi}, e^{j4\pi f/\phi}, \cdots]$, ϕ is the scanning rate on camera, and V_n is the noise subspace in Equation 5. Due to the impact from background noise, the estimated frequency f from Equation 6 may not exactly match the frequencies defined in MFSK
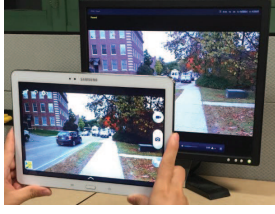
Fig. 5. Experimental Setup

demodulation scheme, sometimes even beyond the spectrum range (i.e, $7Hz \sim 14Hz$). Therefore, a calibration method is developed to correct the falsely estimated frequencies as follows:

$$\widehat{f} = \begin{cases} [f] & ,7Hz \leq f \leq 14Hz \\ 7Hz & ,f < 7Hz \\ 14Hz & ,f > 14Hz, \end{cases} \quad (7)$$

where $\widehat{f}$ is the calibrated frequency and $[\cdot]$ means round operation. Finally the demodulation process is completed by mapping $\widehat{f}$ back to the corresponding bit segment defined in Table III.

V. PERFORMANCE EVALUATION

In this section, we examine the performance of Uber-in-light. Particularly, we study the user experience on watching our modulated videos, and evaluate the accuracy and throughput using the published dataset from HiLight [6]. To facilitate the performance evaluation, we build the prototype to inspect the proposed system under the impact from various environmental factors, including grid size (i.e, the number of pixels in each grid, and the unit is kilopixel Kp), video content, distance between camera and screen, screen size, and observation angle. We first describe the prototype implementation of Uber-in-light and then proceed to the experiments.

A. Prototype Implementation and Evaluation Metrics

Prototype Implementation. Uber-in-light prototype consists of transmitter and receiver implemented in C++ integrated with OpenCV, which is widely adopted computer vision library. The transmitter takes both bit stream and video clip as inputs, generates the G-Channel synchronization signal and RB-Channel complementary data, and then plays back the multiplexed video stream at a specified frame rate. The receiver captures the frames, locates the transmitter screen and divides it into grids, extracts the color intensity changes, and decodes the recovered frames into output data. For real-time transmission, Uber-in-light leverages multi-threaded CPU implementation on the main components. The transmitter utilizes OpenCV for frame processing (i.e., modulation, coding, etc.) and in the meanwhile creates OpenGL thread for real-time display. Particularly, our implementation supports real-time playback on screen at $60FPS$ frame rate in our experimental settings. In the meanwhile, the receiver captures the video displayed on screen via the embedded camera on tablet or smartphone and performs data demodulation concurrently.

Evaluation Metrics. To comprehensively evaluate the system performance, we define three metrics: *perception score*, *accuracy* and *throughput*. Specifically, *perception score* is used to evaluate the subjective perception quality when users watch the videos multiplexed with unobtrusive data transmission, and we use the same 5 levels of perception score defined

in Table I. *Accuracy* is defined as the percentage of error-free received bits over all bits transmitted over screen-camera channel, while *throughput* is defined as the amount of data that are correctly recovered at the receiver.

B. Experimental Evaluation

Experimental Setup. As shown in Figure 5, the experiments are carried out with the following default settings (unless otherwise mentioned). Transmitter is Lenovo 19 inches LCD screen with 1600×900 resolution and $60Hz$ refresh rate. Receiver is Samsung Note 10.1 with rear camera $5Mp$ recording videos at the scanning rate $30FPS$. All experiments were carried out in typical in-door office settings at the default distance between transmitter and receiver $40cm$. Further, we generate a random bits stream as the input data, and apply it to all experiments to ensure the consistency across different tests. The input data is modulated as $8FSK$ signal with the modulation rate $600ms$ per bit segment. All the videos, where the input data is embedded, are played with $720p$ HD resolution in full screen mode, and the test on each of videos is repeated 10 times. As a comparison, we also implement HiLight system based on the demo code[1] published in GitHub with exactly the same parameters in [6]. We do not compare with InFrame++ [7] because the high-end monitor with high frame rate (i.e., $120FPS$) deployed in their system is not commonly available in everyday life.

User Experience Study. We first conduct user experience study to examine whether Uber-in-light causes any noticeable visual artifacts on the original video content. Our study involves 12 participants aged between 18 to 35. We select the 20 videos (each one lasts 30 seconds) covering different types of video contents, and create an online survey[2] to assess the user experience on differencing the original video and modified video where the unobtrusive data transmission embedded. The survey includes 4 categories of tests: 1) Different grid sizes; 2) Different color intensity changes δ; 3) Different types of video contents; and 4) Different video resolutions.

The average perception scores among all the participants is presented in Table IV. Based on the results of user experience study, we find that: 1) smaller color intensity change δ is more unobtrusive to the human eyes, which is within our expectation. For example, the average perception score is close to 5 given $\delta = 0.8$. Then we set $\delta = 0.8$ as the default value in our system; 2) smaller grid size allows more changes to be imperceptible by human eyes. The grid edges are the most perceptive locations due to the luminance difference between adjacent grids, but the edges of smaller grid is easy to be neglected by human eyes within the original video content (e.g., 4.8 for grid size $2.4Kp$ while 3.7 for $240Kp$). The

TABLE IV
USER EXPERIENCE STUDY

Grid Size	$240Kp$	$12Kp$	$4Kp$	$2.4Kp$
Perception Score	3.7	4.3	4.5	4.8
δ	0.4	0.8	1.2	1.6
Perception Score	4.8	4.4	3.5	2.7
Video Content	Static	Drama	Scenery	Sports
Perception Score	4.7	4.2	4.8	4.8
Resolution	480p	720p	1080p	
Perception Score	4.6	4.6	4.8	

[1]https://github.com/Tianxing-Dartmouth/HiLight
[2]http://goo.gl/forms/TTiVmKunih

(a) Accuracy.

(b) Throughput.
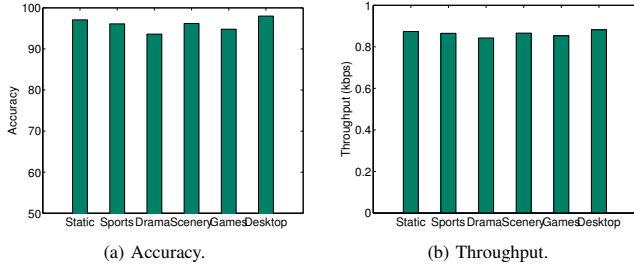
Fig. 6. Performance of different types of video contents.

(a) Static scene.

(b) Dynamic scene.

Fig. 8. Throughput with different grid sizes.

(a) Static scene.

(b) Dynamic scene.
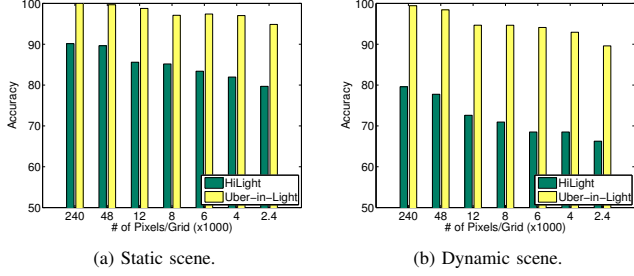
Fig. 7. Accuracy with different grid sizes.

(a) Accuracy.

(b) Throughput.

Fig. 9. Performance with different distances between screen and camera.

default grid size in following experiments is chosen as $12Kp$; 3) modified luminance changes are not obvious in highly dynamic videos. Human eyes are less sensitive to the modified luminance changes if the original video content already has more drastic luminance changes. In our experiment, sports video has higher perception score (i.e., 4.8) than all the others types due to the its highly dynamic content; 4) video resolution does not affect much on user experience. The average perception score maintains above 4.6 when the video resolution increases from $480p$ to $1080p$. Hence, $720p$ is set as the default video resolution in the experiments.

Impact of different types of video contents. We study the performance of Uber-in-light when displaying different types of videos on screen. Particularly, 6 different types of video contents are tested, including both static (i.e., image) and dynamic scenes(i.e., video clips of drama, sports, scenery, game and desktop activities), which cover a wide range of color intensity and frame transitions. The evaluation results are averaged across 20 different clips of the same type of video, each lasting for $1min$. Figure 6 gives the average data accuracy and throughput with different types of video contents. Overall, our system always maintains above 94% accuracy and $0.85Kbps$ throughput for different types of video contents with default grid partition on screen. Even for the sports and game videos that involve lots of abrupt scene changes, the differences from other videos are within 5% and $0.04Kbps$ for accuracy and throughput, respectively. The above results confirm that our proposed system is generic to different types of video contents on ensuring unobtrusive data transmission with high accuracy and throughput.

Impact of grid size. We next study how the grid size affects our system performance. Particularly, we vary the grid size from $240Kp$ to $2.4Kp$ per grid on the screen, and test 20 different videos covering both static and dynamic(i.e., including different types of video contents) scenes. Given a particular screen, fewer pixels are involved in each grid as the grid size decreases, so the accumulated color intensity changes correspondingly decreases. It makes the data transmission more susceptible to attenuation and interference, thereby
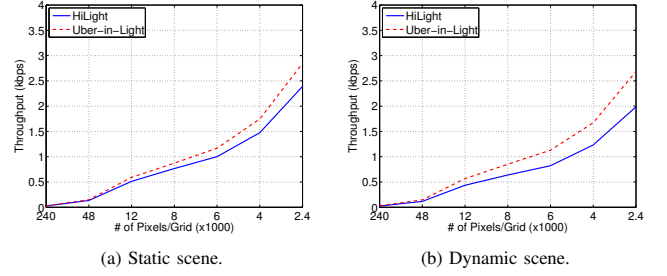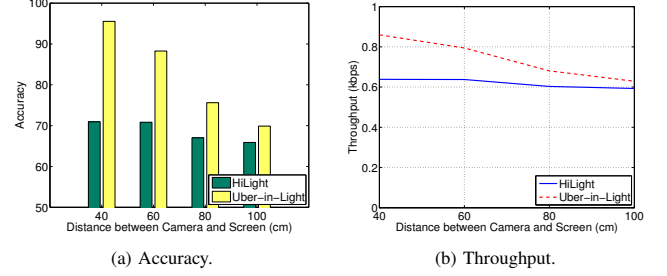
resulting in accuracy degradation. On the other side, more parallel channels (i.e., each grid represents one independent transmitting unit) are involved as the grid size decreases, so the overall throughput still increases though the accuracy on each channel decreases. We compare the performance between Uber-in-light and HiLight under identical transmission rate in Figures 7 and 8. For static scene test in Figures 7(a) and 8(a), Uber-in-light consistently outperforms HiLight with at least 10% accuracy improvement, and the throughput improvement becomes larger as the grid size decreases. Particularly, Uber-in-light achieves over 95% accuracy with small grid size $2.4Kp$ but HiLight only has 67% accuracy (This leads to $2.77Kbps$ throughput in Uber-in-light vs. $2.38Kbps$ in Hi-Light). For dynamic scene test in Figures 7(b) and 8(b), the accuracy and throughput improvement (20% and $0.79Kbps$) of Uber-in-light become even larger than those in static scene test, so our system is more robust to dynamic scenes.

Impact of distance between screen and camera. In the following paragraph, we discuss the impact of distance between screen and camera. Given fixed position of screen, we gradually increase the distance between screen and camera from $40cm$ cm to $100cm$ in the experiment. The average accuracy and throughput is provided in Figure 9. The longer the distance between screen and camera, the more severe the attenuation and inference the emitted visible light signal will experience. Therefore, as the distance increases, the accuracy of Uber-in-light decreases from 96% to around 70% while the throughput decreases from $0.86Kbps$ to $0.63Kbps$. But Uber-in-light still outperforms HiLight with at least 6% accuracy and throughput improvement within $100cm$. Furthermore, the performance of HiLight is relatively stable as the distance increases but with lower accuracy and throughput, so Uber-in-light is more suitable for near field communication.

Impact of observation angle. To examine the impact of the observation angles, we conduct the experiment by rotating the camera horizontally by 0 to 60 degrees with respect to fixed screen while keeping the in-between distance at $40cm$. As the results shown in Figure 10(a), we observe that the accuracy keeps decreasing as the observation angle increases, but it still

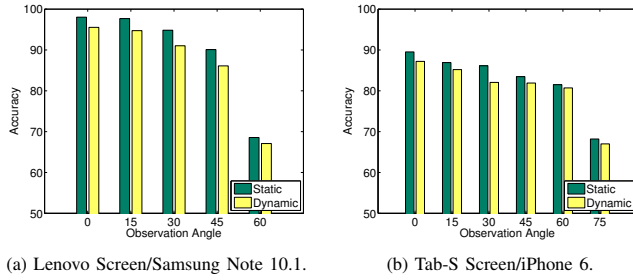(a) Lenovo Screen/Samsung Note 10.1. (b) Tab-S Screen/iPhone 6.

Fig. 10. Accuracy with different observation angles.

achieve over 87% accuracy with the observation angle as wide as 45 degrees. Once the observation angle reaches 60 degrees, the accuracy sharply degrades to below 70%. This is because wide observation angle makes the camera difficult to identify the color intensity changes due to the restricted propagation direction of light source on screen. As a comparison, we repeat the test with Samsung Tab-S screen and IPhone 6 as the transmitter and receiver. Figure 10(b) finds similar varying tend on the accuracy with new transceiver pair, but the overall accuracy becomes worse due to smaller Tablet screen adopted in the experiment. Further, since IPhone 6 is equipped with high-sensitivity camera, the accuracy drops below 70% until the observation angle reaches 75 degrees.

Impact of screen size. Finally, we study the impact of screen size to our system. We choose iPhone 6 as the receiver to test with the screens of different sizes varying from 5 to 27 inches. The reason why we choose IPhone 6 is to ensure the camera could capture the whole screen of large size (e.g., 27 inches). The accuracy and throughput results are listed in Table V. It is obvious to find that, larger screen size with higher resolution results in higher accuracy and throughput. In particular, we note that the maximum throughput reported in HiLight [6] is only $0.6Kbps$ with the same Mac screen and IPhone 6 camera, while Uber-in-light can achieve $1.1Kbps$ even with smaller screen of Samsung Note 4 while maintaining the accuracy above 90%. The above results validates that our system indeed achieves high accuracy and high throughput for unobtrusive visible light communication.

VI. CONCLUSIONS

In this paper, we present a novel unobtrusive visible light communication system called *Uber-in-light* that is applicable to any screen and camera devices. Specifically, Uber-in-light encodes the data as complementary color intensity changes across different color channels on screen, which could be successfully decoded by camera while leaving the human visual perception unaffected. An MFSK modulation scheme is designed with dedicated orthogonal synchronization signal

TABLE V
THROUGHPUT WHEN ACHIEVING 90% ACCURACY ON SCREENS OF DIFFERENT SIZES

Screen	Grid Size	Content	Accuracy	Throughput
27" 2560x1440	600	Static	90	5.4
MAC monitor	240	Dynamic	90	2.2
22" 1920x1200	600	Static	90	5.4
Dell monitor	240	Dynamic	92	2.2
15.4" 1920x1080	360	Static	90	3.3
HP ENVY laptop	240	Dynamic	90	2.2
10" 2560x1600	240	Static	90	2.2
Samsung Tab S	180	Dynamic	90	1.6
5.7" 2560x1440	180	Static	91	1.6
Samsung Note 4	120	Dynamic	91	1.1

to endow the screen-camera channel with high throughput. We also develop an enhanced MUSIC-based demodulation scheme to ensure highly accurate data transmission. The proposed unobtrusiveness design is validated on different types of video content and various resolutions with real implementation. Extensive real-time experiments confirm the effectiveness of the real prototype implementation of Uber-in-light under the impact of different system and environmental factors.

REFERENCES

[1] B. Zhang, K. Ren, G. Xing, X. Fu, and C. Wang, "Sbvlc: Secure barcode-based visible light communication for smartphones," in *INFOCOM*. IEEE, 2014, pp. 2661–2669.

[2] W. Hu, H. Gu, and Q. Pu, "Lightsync: Unsynchronized visual communication over screen-camera links," in *MobiCom*. ACM, 2013, pp. 15–26.

[3] ISO, "Automatic identification and data capture techniques - qr code 2005 bar code symbology specificatio," International Organization for Standardization, ISO ISO/IEC 18004:2006, 2006.

[4] W. Huang and W. H. Mow, "Picode: 2d barcode with embedded picture and vicode: 3d barcode with embedded video," in *Mobicom*. New York, NY, USA: ACM, 2013, pp. 139–142.

[5] G. Woo, A. Lippman, and R. Raskar, "Vrcodes: Unobtrusive and active visual codes for interaction by exploiting rolling shutter," in *ISMAR*, Nov 2012, pp. 59–64.

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

[7] A. Wang, Z. Li, C. Peng, G. Shen, G. Fang, and B. Zeng, "Inframe++: Achieve simultaneous screen-human viewing and hidden screen-camera communication," in *MobiSys*. ACM, 2015, pp. 181–195.

[8] A. Cheddad, J. Condell, K. Curran, and P. Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal processing*, vol. 90, no. 3, pp. 727–752, 2010.

[9] A. Burton, H. Le Minh, Z. Ghassemlooy, S. Rajbhandari, and P. Haigh, "Performance analysis for 180 receiver in visible light communications," in *ICCE*. IEEE, 2012, pp. 48–53.

[10] S. Schmid, G. Corbellini, S. Mangold, and T. Gross, "An led-to-led visible light communication system with software-based synchronization," in *GC Wkshps, 2012 IEEE*, Dec 2012, pp. 1264–1268.

[11] Z. Li, W. Chen, C. Li, M. Li, X.-Y. Li, and Y. Liu, "Flight: Clock calibration using fluorescent lighting," in *ACM MobiCom*, 2012, pp. 329–340.

[12] R. LiKamWa, D. Ramirez, and J. Holloway, "Styrofoam: A tightly packed coding scheme for camera-based visible light communication," in *VLCS*. ACM, 2014, pp. 27–32.

[13] N. Rajagopal, P. Lazik, and A. Rowe, "Visual light landmarks for mobile devices," in *IPSN*. IEEE Press, 2014, pp. 249–260.

[14] T. Hao, R. Zhou, and G. Xing, "Cobra: color barcode streaming for smartphone systems," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 2012, pp. 85–98.

[15] S. D. Perli, N. Ahmed, and D. Katabi, "Pixnet: interference-free wireless links using lcd-camera pairs," in *MobiCom*. ACM, 2010, pp. 137–148.

[16] W. Hu, J. Mao, Z. Huang, Y. Xue, J. She, K. Bian, and G. Shen, "Strata: layered coding for scalable visual communication," in *MobiCom*. ACM, 2014, pp. 79–90.

[17] A. Mohan, G. Woo, S. Hiura, Q. Smithwick, and R. Raskar, "Bokode: imperceptible visual tags for camera based interaction from a distance," *TOG*, vol. 28, no. 3, p. 98, 2009.

[18] F. A. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding-a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[19] Y. Zhang, "Digital watermarking technology: A review," in *FCC*. IEEE, 2009, pp. 250–252.

[20] N. Carlin, "Another interpretation of the data from blackwell, hr (1946): Contrast thresholds of the human eye," 1997.

[21] E. Reinhard, E. A. Khan, A. O. Akyuz, and G. Johnson, *Color imaging: fundamentals and applications*. CRC Press, 2008.

[22] Wikipedia, "Relative luminance," [Online; accessed 12-July-2015]. [Online]. Available: https://en.wikipedia.org/wiki/Relative_luminance

[23] G. Bradski, "Opencv," *Dr. Dobb's Journal of Software Tools*, 2000.

[24] M. H. Hayes, *Statistical digital signal processing and modeling*. John Wiley & Sons, 2009.

[25] D. Kundu, "Modified music algorithm for estimating doa of signals," *Signal processing*, vol. 48, no. 1, pp. 85–90, 1996.

[26] Z. Shan and T.-S. P. Yum, "A conjugate augmented approach to direction-of-arrival estimation," *IEEE SP*, vol. 53, no. 11, pp. 4104–4109, 2005.