

Concatenated Channel Coding with Ring Structure in Screen-Camera Links

Tzu-Hao Chang, Hsu-Feng Hsiao

Department of Computer Science,

National Yang Ming Chiao Tung University, Hsinchu, Taiwan

Abstract—This paper presents a novel approach to screen-camera communication, addressing the growing demand for efficient data transmission in visible light communication systems. We introduce Concatenated Channel coding with Ring structure for unequal error protection (CCRCode), a robust coding scheme designed to overcome the challenges of screen-camera links. CCRCode leverages a two-tier error correction strategy: an intra-symbol level using Low Density Parity Check (LDPC) codes for data correctness, and an inter-symbol level employing a unique ring-structured layer-aligned multipriority rateless codes with additional LDPC codes for data integrity and unequal error protection. Our method outperforms existing techniques in handling heterogeneous data with varying importance levels. We develop an optimization model that dynamically adjusts coding parameters based on channel conditions, ensuring optimal performance across diverse scenarios. Extensive experiments demonstrate CCRCode’s superior efficacy in terms of recovery rate and utility under various challenging conditions, including different distances, viewing angles, and screen shifts. Notably, CCRCode achieves complete data recovery in half the time required by competing methods.

I. Introduction

Visible Light Communication (VLC) is transforming wireless communication by leveraging ubiquitous LED lighting and camera-equipped smart devices. Unlike traditional radio frequency (RF) systems, VLC offers inherent security advantages due to the directionality of optical transmission [1]. A compelling VLC application is using two-dimensional (2D) barcodes for information exchange between display screens and smartphone cameras, enabling infrastructure-free data transfer where conventional wireless connections are unavailable or security is critical.

Advanced 2D barcode designs like Quick Response (QR) codes [2] and Microsoft’s High Capacity Color Barcodes (HCCB) [3] have expanded the information capacity of optical codes. However, these barcodes primarily transmit URLs, requiring a separate connection to access the actual content. VLC-based barcode streaming can transcend these limitations by displaying a continuous stream of color-modulated barcodes on the sender’s screen, captured and decoded by the receiver’s camera, enabling direct transmission of substantial data payloads [4] [5] [6] [7] [8].

Recent advancements position VLC as a promising solution for 5G and Internet of Things (IoT) challenges. [9] reviews VLC’s prospects, approaches, and future directions in IoT applications, highlighting its potential for

massive connectivity. [10] proposes a practical vehicle-to-vehicle VLC propagation model considering outdoor propagation characteristics. Machine learning has also enhanced screen-camera communication, with DeepLight [11] using deep neural networks for robust, imperceptible data transmission and DeepCCB-OCC [12] employing convolutional neural networks for accurate symbol identification.

Despite its potential, barcode streaming over screen-camera links faces challenges such as ambient light, and the relative distance and angle between screen and camera, impacting the receiver’s ability to accurately detect and decode color symbols. Diverse camera performance characteristics introduce varying errors and erasures, necessitating error correction. Furthermore, the variation in data priority within transmitted content, such as images, videos, and composite files, requires unequal error protection (UEP) to ensure successful transmission of the most critical information.

To address these challenges, we propose CCRCode, a novel Concatenated Channel coding scheme with Ring structure for UEP. Our two-tier approach employs Low Density Parity Check (LDPC) codes for intra-symbol data correctness and a proposed ring-structured variant of Layer-aligned Multipriority Rateless Codes (LMRC) with LDPC for inter-symbol erasure resilience and unequal data protection. Furthermore, we develop an intelligent optimization framework that dynamically adapts the coding parameters based on real-time channel conditions, maximizing the utility and recovery rate of the transmitted data.

Extensive experiments demonstrate CCRCode’s superior performance under diverse scenarios, achieving complete data recovery in half the time of other methods. The remaining sections detail CCRCode’s system architecture, channel coding design, optimization, experimental results, and conclusions.

II. Methodology

This section presents the architecture and principles of our proposed Concatenated Channel coding with Ring structure for Unequal Error Protection (CCRCode), designed to address the challenges of screen-camera links while providing robust error correction and prioritized data protection.

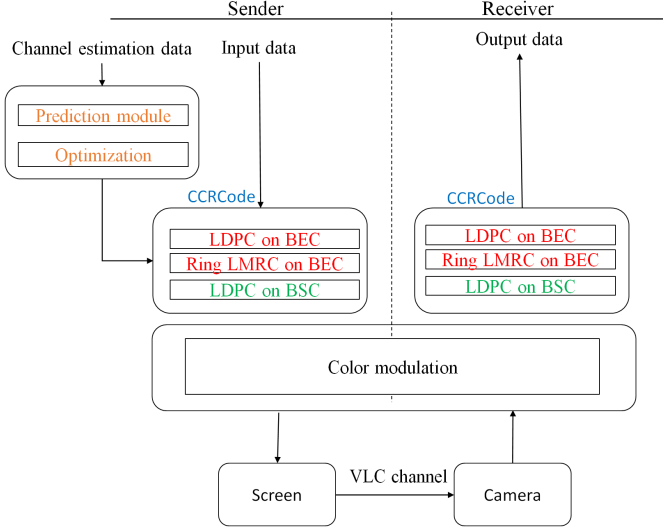


Fig. 1. The system architecture of CCRCODE, illustrating the interplay between sender and receiver modules.

A. System Architecture Overview

Fig. 1 illustrates our CCRCODE system architecture. The sender captures designated barcodes from the receiver, using channel feedback to dynamically optimize error correction parameters. The original data is then encoded using our concatenated channel coding scheme, color modulated, and cyclically displayed as multi-colored video barcodes.

The receiver records and locates the barcodes, extracting data through color demodulation. Error correction decoders process the received data to recover the original information, utilizing RDCODE [4] for color modulation.

Fig. 2 depicts our concatenated channel coding data flow. Original data, clustered by importance, forms input symbol sets $I_{i,j}$ for each layer j within section i . These symbols undergo LDPC encoding (C_j) at the inter-symbol level, with $k_{inter,j}$ and $n_{inter,j}$ representing input and intermediate symbol lengths for layer j in a section. The encoded intermediate symbols are processed by our proposed Ring LMRC, yielding $n(1+\gamma) = k(1+\epsilon)$ output symbols, where γ and ϵ are overheads of Ring LMRC and inter-symbol level, respectively. Each output symbol is LDPC-encoded at the intra-symbol level, creating η -bit final symbols. The final symbols are color modulated into barcode blocks and assembled into frames for display.

B. Concatenated Channel Coding with Ring Structure

To address the dual challenges of channel noise and uncaptured barcodes in screen-camera links, we propose a two-tier concatenated channel coding scheme. The intra-symbol level focuses on data correctness, while the inter-symbol level ensures data integrity.

1) Intra-Symbol Level Error Correction: At the intra-symbol level, we employ Low-Density Parity-Check

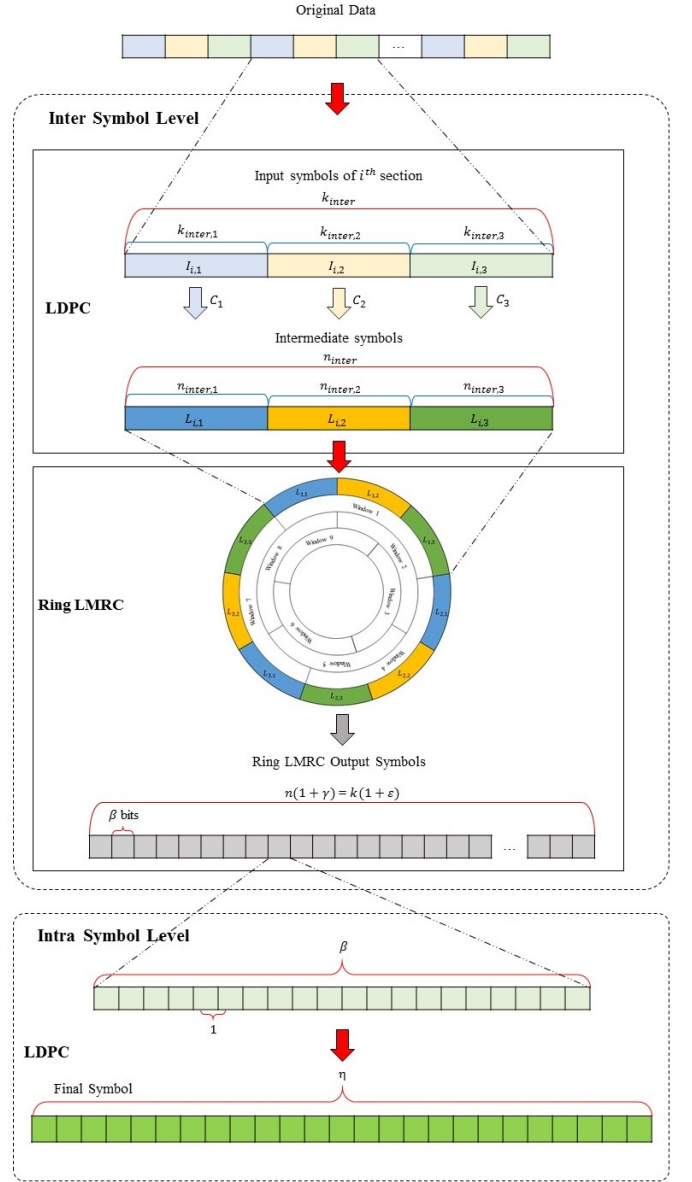


Fig. 2. Data flow of the proposed concatenated channel coding scheme, showcasing the interplay between intra-symbol and inter-symbol level error correction.

(LDPC) codes [13] with Gallager's decoding algorithm to mitigate channel noise errors. LDPC codes provide robust verification of data correctness through matrix multiplication. If decoding of a received codeword fails, it is marked as an erasure for handling by the inter-symbol level. This effectively converts the channel from a combined Binary Symmetric Channel (BSC) and Binary Erasure Channel (BEC) to a pure BEC.

2) Inter-Symbol Level Error Correction with Ring-Structured Rateless Codes: The inter-symbol level recovers invalid output symbols using a combination of LDPC and the proposed Ring-structured Layer-aligned MultiPriority Rateless Codes (Ring LMRC), inspired by

Raptor codes [14]. This synergy of pre-code and rateless code enables Unequal Error Protection (UEP) for data of varying importance.

While the original LMRC [15] excels with layered data like video streams, it can struggle with smaller data volumes due to uneven sliding window coverage. Recognizing the cyclical nature of video barcodes, we reshape LMRC into a ring structure (Fig. 2). The terminal windows now wrap around to the initial data, ensuring uniform coverage and equitable protection within the same priority level.

As depicted in Fig. 2, the inter-symbol level begins with LDPC encoding, which increases the input symbol size. For k_{inter} input symbols per section and an LDPC code rate $R_{inter} \in (0, 1)$, the resulting number of intermediate symbols is $n_{inter} = k_{inter}/R_{inter}$. Each N -layer section of input symbols, denoted as $(I_{s,1}, I_{s,2}, \dots, I_{s,N})$, has a length of $k_{inter,i}$, with the total length given by $k_{inter} = \sum_{i=1}^N k_{inter,i}$. To achieve N -level UEP, we use N distinct LDPC codes $(C_1, C_2, \dots, C_N)$ with rates $(R_{inter,1}, R_{inter,2}, \dots, R_{inter,N})$. This pre-coding step increases the data size by a factor of $\sum_{i=1}^N \alpha_i / R_{inter,i}$, where $\alpha_i = k_{inter,i} / k_{inter}$ represents the size ratio of the i^{th} layer, and $R_{inter,i} = k_{inter,i} / n_{inter,i}$ denotes the LDPC code rate for the i^{th} inter-symbol layer.

Subsequently, Ring LMRC is applied to the intermediate symbols, ensuring uniform window coverage for each data segment. To generate an output symbol, Ring LMRC selects a degree d according to a predefined distribution and chooses d intermediate symbols from the N layers based on the weight parameters $(\omega_1, \omega_2, \dots, \omega_N)$, where the probability of selecting a symbol from layer i is proportional to ω_i . The output symbol is formed by XORing the selected intermediate symbols, as in [15]. The generated output symbols then advance to the intra-symbol level for further processing.

The intra-symbol level's LDPC code rate, $R_{intra} = \beta/\eta$, plays a dual role: it determines both the recovery capability (i.e., the count of valid output symbols for the inter-symbol level) and the number of original bits per symbol. Consequently, varying R_{intra} alters the original symbol count for a fixed data size. A higher R_{intra} , where $0 < R_{intra} < 1$, yields fewer original symbols. At the inter-symbol level, the received output symbol count is $n(1 + \gamma)$, corresponding to an overhead of $k(1 + \epsilon)$. The relationship between these overheads is governed by $(1 + \epsilon) = (1 + \gamma)/R_{inter}$. By carefully tuning R_{inter} and γ , we can achieve the desired ϵ .

C. Optimization of CCRCode

Modifying $R_{inter,i}$ directly impacts γ , while R_{intra} influences the utilizable symbol count at the inter-symbol level. To optimize system performance over screen-camera links, we can strategically control R_{intra} of LDPC code at intra-symbol level, $R_{inter,i}$ of pre-codes at inter-symbol level, and identify the optimal weight settings ω_i of Ring LMRC for each i^{th} layer.

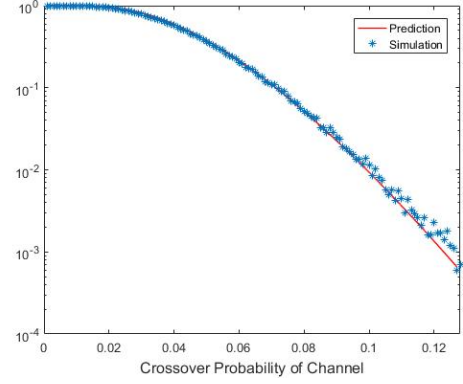


Fig. 3. Validation of the probability prediction for successfully decoded codewords against simulations at code rate=0.5.

Defining an appropriate utility function is crucial for assessing post-decoding performance. Drawing inspiration from the layered structure of image and video data, where enhancement layers refine the quality based on the received base layer, we adopt a similar paradigm for our N -layer data. If the base layer recovery is poor, the higher-layer improvements offer little gain. The utility function $U(\mathbf{p}, \mathbf{k})$ captures this inter-layer dependency:

$$U(\mathbf{p}, \mathbf{k}) = \sum_{l=1}^S \sum_{i=1}^N k_{l,i} \cdot n_{l,i} \cdot (1 - p_{l,i}) \cdot \prod_{j=1}^{i-1} (1 - p_{l,j})^{k_{l,j}}, \quad (1)$$

where $k_{l,i}$ and $n_{l,i}$ represent the data count and information bits of the i^{th} layer in the l^{th} section, respectively; $p_{l,i}$ denotes the erasure rate of the i^{th} layer in the l^{th} section; and vectors $\mathbf{p}$ and $\mathbf{k}$ contain the final erasure rates and data sizes for each layer across all sections. Equation (1) captures the notion that the i^{th} layer contributes only when all preceding layers ($1 \leq j \leq i-1$) are perfectly recovered.

To gauge channel conditions, our prediction module estimates the error rate ϵ_i for each of the $c \cdot h$ blocks in a barcode, where $i = 1, \dots, c \cdot h$. By capturing the receiver's unique barcode prior to encoding, the sender obtains the error rate vector ϵ . The objective is to find the optimal parameter set $P_\epsilon = \{R_{intra}, R_{inter,1}, R_{inter,2}, \dots, R_{inter,N}, \omega_1, \omega_2, \dots, \omega_N\}$ that maximizes Equation (1) for the given ϵ , subject to $R_{intra} \in \{R_1^{intra}, R_2^{intra}, \dots, R_m^{intra}\}$ and $R_{inter,i} \in \{R_1^{inter}, R_2^{inter}, \dots, R_m^{inter}\}$.

For a codeword with n_{intra} coding elements, the probability of successful decoding, $P_{success}$, can be predicted using the analysis tool based on a technique similar to that presented in [16]:

$$P_{success} = \sum_i \Pr(e_i) (1 - P_{b|i})^{n_{intra}}, \quad (2)$$

where $P_{b|i}$ is calculated according to [16], and $\Pr(e_i)$ represents the probability of encountering error e_i . Fig. 3

validates the accuracy of Equation (2) against simulations. By predicting the successfully decoded symbol count (i.e., the inter-symbol level overhead ϵ) for a fixed barcode count and R_{intra} , and leveraging the sender's knowledge of the total barcode count, we obtain the overhead ϵ_j corresponding to the code rate R_j^{intra} .

The final step involves an exhaustive search over $R_{inter,i}$ and ω_i using prediction modules similar to those proposed in [16] and [15], respectively, to maximize the utility function (Equation (1)) under the overhead ϵ_j . The sender then encodes the data using the optimal parameter set P_ϵ and transmits it to the receiver. In cases where multiple P_ϵ yield the same utility, we favor the lower R_j^{intra} value.

III. Experiments

We implemented our system as an extension of RDCode [4] using Scala for Android platforms. For each experiment, we conducted 20 tests with original data clustered into three importance levels, with the ratio of data sizes for the three layers being $k_{inter,1} : k_{inter,2} : k_{inter,3} = 1 : 3 : 6$.

We compare our method with RDCode using different code rates R at the block level, denoted as RDCode(R). RDCode employs tri-level error correction. We also implemented LDPC codes with code rate R at the block level on top of RDCode, denoted as Mod. RDCode(R). Following RDCode's approach, we place more important data in positions closer to the center of the barcode. The parameters for RDCode and Mod. RDCode are shown in Table I. Additionally, we implemented LMRC code as the rateless code at our inter-symbol level, denoted as LMRC. Each method incorporates a prediction module, allowing RDCode and Mod. RDCode to accurately determine the locations of redundant blocks and place more important data into blocks with lower error rates.

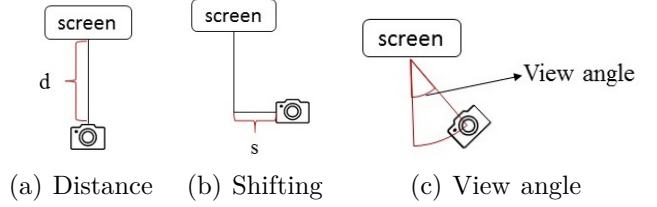


Fig. 6. Experimental setups for evaluating distance, shifting, and viewing angle effects.

We measure the recovery rate (ratio of successfully decoded data to total data) and utility, calculated using Equation (1), after the receiver has decoded the data. The experimental settings for different distances, shifts, and view angles are illustrated in Fig. 6.

TABLE I
Parameters of RDCodes and Mod. RDCode

Method(code rate)	Frame level (p)	Packet level (q)
RDCode(0.5)	26	3
Mod. RDCode(0.5)	26	3
RDCode(0.6)	36	2
Mod. RDCode(0.6)	36	2
RDCode(0.7)	36	2
Mod. RDCode(0.7)	36	2

A. Impact of Distance

Fig. 6(a) and 4 illustrate the receiver placement and error distribution of received barcodes at different distances d . Blocks denoted in red could not be captured by the receiver, except for the center locator block.

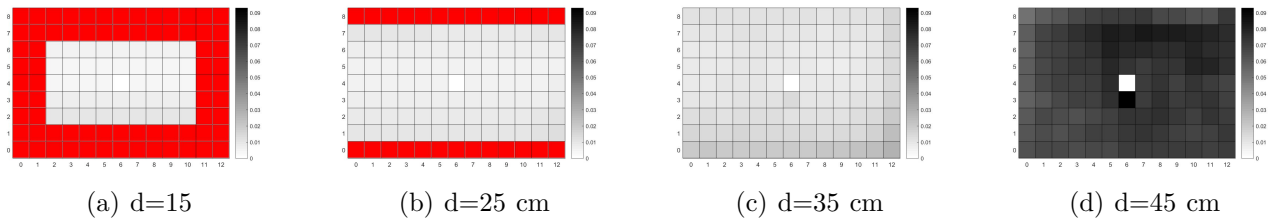


Fig. 4. Error distribution patterns at varying transmitter-receiver distances. Red blocks indicate areas where data capture failed.

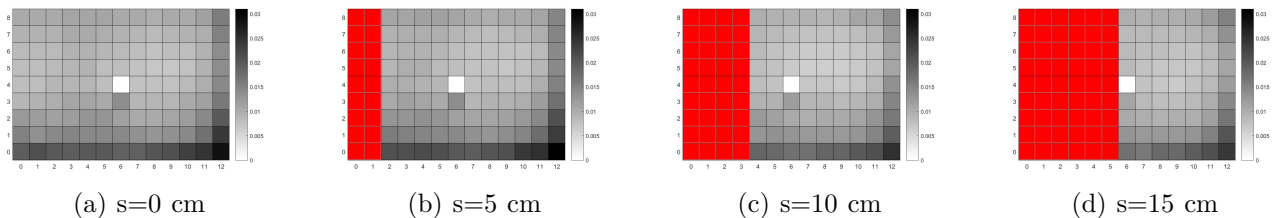


Fig. 5. Error distribution patterns under various screen shifting conditions. Red blocks indicate areas where the receiver failed to capture data.

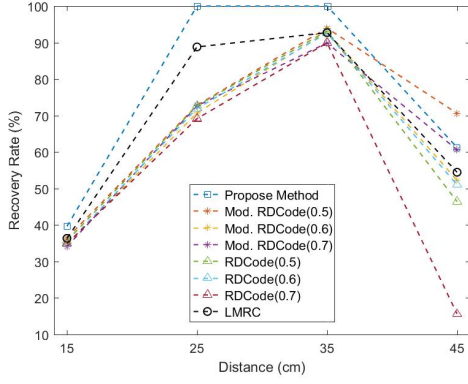


Fig. 7. Recovery rates across varying transmitter-receiver distances.

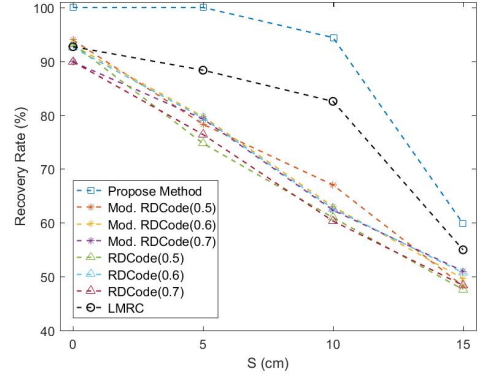


Fig. 9. Recovery rates under varying degrees of screen shifting.

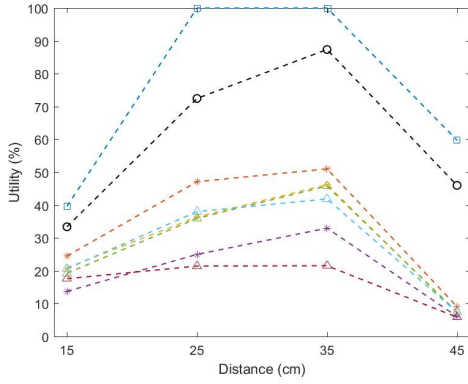


Fig. 8. Utility across varying transmitter-receiver distances.

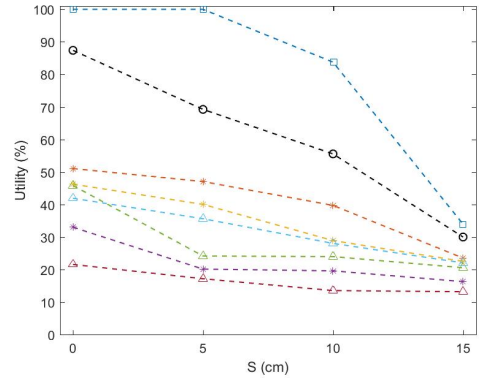


Fig. 10. Utility under varying degrees of screen shifting.

As shown in Fig. 4, smaller d values result in lower error severity and a smaller capturable portion of the barcode for $d \leq 35$ cm. At $d = 35$ cm, the receiver can capture the entire barcode with the lowest error severity. At $d = 45$ cm, the system encounters more serious errors compared to other distances.

The experimental results are shown in Fig. 7 and 8. In Fig. 8, our proposed method demonstrates better utility than other methods, even when the recovery rate is lower than Mod. RDCode(0.5) at $d = 45$ cm. Due to unequal error protection, our method prioritizes the recovery of more important data over other layers, even when most data is not received or encounters serious channel errors. Additionally, Ring LMRC outperforms LMRC over screen-camera links.

B. Effect of Screen Shifting

Figures 6(b) and 5 illustrate the receiver placement and error distribution of received barcodes at different shifts s . As shown in Fig. 5, larger s values result in a smaller capturable portion of the barcode. Each shift loses 15.5% of the barcode. The entire barcode can be captured at $s = 0$ cm.

The experimental results are shown in Fig. 9 and 10. Our proposed method achieves the best performance. By leveraging unequal error protection, we sacrifice less important data to recover as much important data as possible.

C. Influence of Viewing Angle

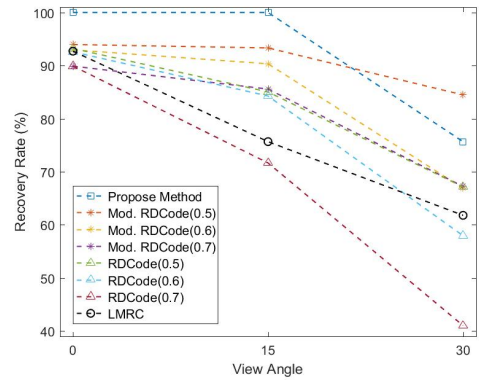


Fig. 11. Recovery rates across different viewing angles.

Fig. 6(c) shows the receiver's view angle. Even with varying error rates across barcode blocks, our proposed

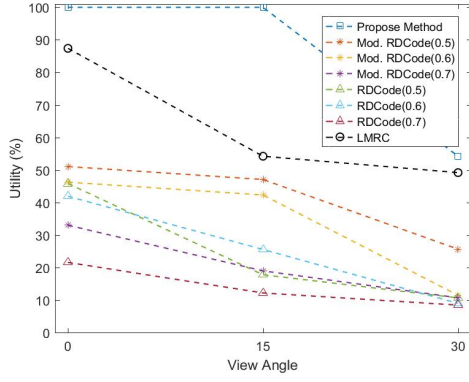


Fig. 12. Utility across different viewing angles.

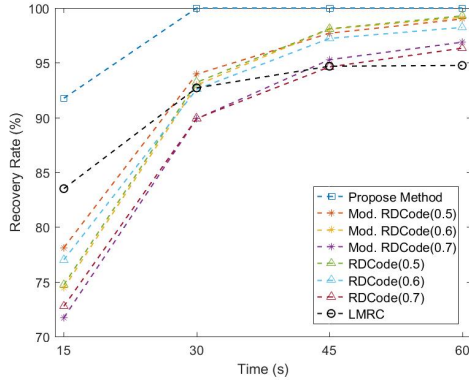


Fig. 13. Recovery rates across different transmission durations.

method achieves better utility than other methods by sacrificing the recovery of less important layers to prioritize the recovery of more important data. Under RDCode and Mod. RDCode, the systematic property holds at the frame and packet levels, as shown in Fig. 11 and 12. Consequently, if more important data has serious errors, it cannot be used to recover other data within the same layers.

D. Transmission Duration Analysis

Fig. 13 shows the recovery rate of each method with respect to different receiving durations. When the system has only 15 seconds of receiving time, none of the methods perform well. Our method is able to recover all data within 30 seconds. However, the other methods cannot recover all data, even when given twice as much time.

IV. Conclusion

This paper introduces CCRCCode, a novel concatenated channel coding scheme with ring structure for unequal error protection in screen-camera communication. Our two-tier strategy, combining LDPC codes and the proposed Ring LMRC, effectively addresses the unique challenges of these links. Furthermore, our optimization framework dynamically adapts coding parameters based on real-time

channel conditions, maximizing the utility and recovery rate of the transmitted data. Extensive experimental evaluations demonstrate the superior performance of CCRCCode across diverse scenarios, including variations in distance, viewing angle, and screen misalignment.

References

- [1] P. H. Pathak, X. Feng, P. Hu, and P. Mohapatra, "Visible light communication, networking, and sensing: A survey, potential and challenges," *IEEE communications surveys & tutorials*, vol. 17, no. 4, pp. 2047–2077, 2015.
- [2] ISO/IEC Standard, "Information technology – automatic identification and data capture techniques – qr code 2005 bar code symbology specification," ISO/IEC, Standard 18004:2015, 2015.
- [3] G. Jancke, "High capacity color barcodes (hccb)," 12 2007, [Online; accessed 24-06-2024]. [Online]. Available: <http://www.microsoft.com/en-us/research/project/high-capacity-color-barcodes-hccb>
- [4] A. Wang, S. Ma, C. Hu, J. Huai, C. Peng, and G. Shen, "Enhancing reliability to boost the throughput over screen-camera links," in *Proceedings of the 20th annual international conference on Mobile computing and networking*, 2014, pp. 41–52.
- [5] S. D. Perli, N. Ahmed, and D. Katabi, "Pixnet: Interference-free wireless links using lcd-camera pairs," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, 2010, pp. 137–148.
- [6] 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*, 2012, pp. 85–98.
- [7] W. Hu, H. Gu, and Q. Pu, "Lightsync: Unsynchronized visual communication over screen-camera links," in *Proceedings of the 19th annual international conference on Mobile computing & networking*, 2013, pp. 15–26.
- [8] M. Stafford, A. Rogers, S. Wu, C. Carver, N. S. Artan, and Z. Dong, "Tetris: Smartphone-to-smartphone screen-based visible light communication," in *2017 IEEE 14th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 2017, pp. 570–574.
- [9] S. S. Oyewobi, K. Djouani, and A. M. Kurien, "Visible light communications for internet of things: Prospects and approaches, challenges, solutions and future directions," *Technologies*, vol. 10, no. 1, p. 28, 2022.
- [10] P. Sharda and M. R. Bhatnagar, "Vehicular visible light communication system: Modeling and visualizing critical outdoor propagation characteristics," *IEEE Transactions on Vehicular Technology*, 2023.
- [11] V. Tran, G. Jayatilaka, A. Ashok, and A. Misra, "Deeplight: Robust & unobtrusive real-time screen-camera communication for real-world displays," in *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*, 2021, pp. 238–253.
- [12] M. T. Kim and B. W. Kim, "Deepccb-occ: Deep learning-driven complementary color barcode-based optical camera communications," *Applied Sciences*, vol. 12, no. 21, p. 11239, 2022.
- [13] R. Gallager, "Low-density parity-check codes," *IRE Transactions on information theory*, vol. 8, no. 1, pp. 21–28, 1962.
- [14] A. Shokrollahi, "Raptor codes," *IEEE transactions on information theory*, vol. 52, no. 6, pp. 2551–2567, 2006.
- [15] H.-F. Hsiao and Y.-J. Ciou, "Layer-aligned multipriority rateless codes for layered video streaming," *IEEE transactions on circuits and systems for video technology*, vol. 24, no. 8, pp. 1395–1404, 2014.
- [16] M. Noor-A-Rahim, K. D. Nguyen, and G. Lechner, "Finite length analysis of ldpc codes," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE, 2014, pp. 206–211.