Artistic QR Codes

Ming-Yang Ho*, Yi-Tien Tsai*, and Ching-Ping Pan*

National Taiwan University
kaminyou@cmdm.csie.ntu.edu.tw, d13944024@csie.ntu.edu.tw,
r13944027@csie.ntu.edu.tw

(a) Ming-Yang Ho **(b)** Yi-Tien Tsai **(c)** Ching-Ping Pan

Fig. 1: Styled color QR codes.

Abstract. QR codes are widely used in daily life for encoding URLs, payment information, and other types of data. However, their traditional black-and-white appearance is visually monotonous and limits their integration into aesthetically appealing contexts. Recent approaches have explored artistic QR code generation using deep learning and diffusion models, but these methods are computationally intensive and impractical for real-time or resource-constrained applications. In this work, we present a lightweight and effective framework for stylizing QR codes using traditional image processing techniques, specifically building upon Halftone QR codes. Our method supports color images, customizable box sizes, and saliency-aware module dropping, enabling the generation of visually enhanced yet scannable QR codes. We provide a fully open-source Python implementation and evaluate our approach across multiple devices and datasets. Experimental results show that our method produces visually appealing QR codes while maintaining high readability on common mobile devices. Code is available at: `https://github.com/Kaminyou/Halftone-Color-QR-Codes`.

Keywords: Stylized QR Code · Halftoning · Image Processing

* These authors contributed equally to this work

1 Division of the Work

- Ming-Yang Ho (F11922208): Extension implementation, Code refinement, Project scope and experiment design
- Yi-Tien Tsai (D13944024): Basic implementation, Basic experiment, Video editing
- Ching-Ping Pan (R13944027): Basic implementation, Ablation studies experiment design and discussion

2 Primary Reference

Chu, H. K., Chang, C. S., Lee, R. R., & Mitra, N. J. (2013). Halftone QR codes. *ACM transactions on graphics (TOG), 32*(6), 1-8. [1]

3 Introduction & Motivation

With the widespread adoption of mobile phones, QR codes have become ubiquitous in daily life [2]. They are commonly used for mobile payments [3], restaurant ordering [4], and even linking to audio guides in museums [5]. However, standard QR codes are typically plain black and white, making them visually unappealing.

Artistic QR codes offer a more attractive alternative. Recently, several approaches have been proposed to enhance the visual appeal of QR codes [6–8], including those guided by text prompts [9]. However, most of these methods rely on deep learning [10], particularly diffusion models [11, 12], which are computationally expensive and slow—limiting their practical use.

In this project, we explore traditional image processing techniques to beautify QR codes efficiently. Specifically, our method is based on Halftone QR codes [1], which leverage the structural features of QR code design in combination with halftoning techniques to *embed a style image into a scannable QR code.*

4 Problem Definition

While the methodology proposed in Halftone QR codes [1] can generate visually appealing stylized QR codes, it has several limitations. **First**, it only supports black-and-white style images, making it incapable of embedding color images into the QR code. **Second**, it restricts the module box size to 3, which limits the clarity and detail of more complex image content. **Lastly**, the original work does not provide any executable implementation, significantly hindering its usability in real-world applications.

Our project addresses all of these limitations (see Fig. 1). Specifically, our implementation supports color images, allowing the embedding of full-color content into QR codes. It also enables flexible box size configurations, making it possible to accommodate more complex image embeddings. Finally, we provide a fully open-source Python implementation. Thanks to Python's portability and our modular code design, users can easily customize the system to suit their specific needs.

5 Contributions

In summary, our research makes the following contributions:

- We extend the Halftone QR code method to support color image stylization.
- Our implementation allows flexible box size settings, enabling the embedding of complex images.
- We release our code as fully open source to facilitate further research and development in this area.

6 Algorithm

6.1 Preliminary

QR Code Structure. A standard QR code consists of both data regions and supplementary information (see Fig. 4). For example, it includes position, alignment, and timing patterns, as well as version and format information. These structural regions are essential for QR code decoding and therefore cannot be modified. As a result, only the data and error correction regions are available for embedding a style image.

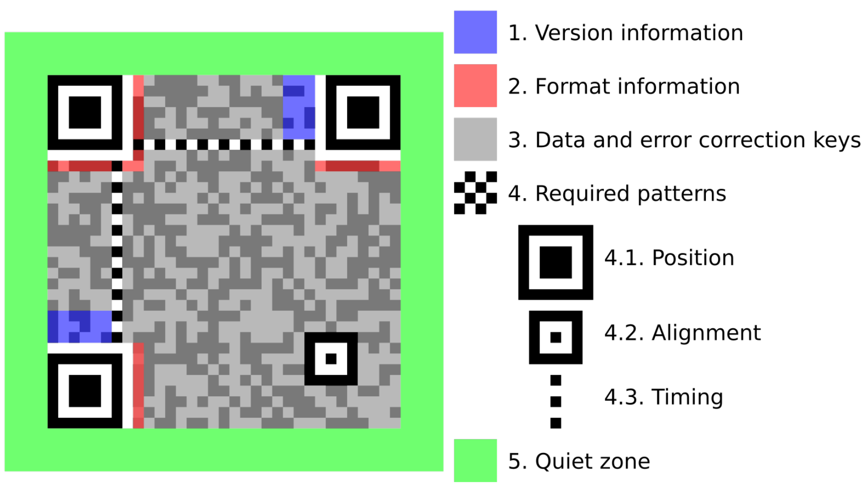

Fig. 2: QR Code Structure. A QR code includes both data and supplementary patterns such as position, alignment, and timing markers. These structural elements are fixed and cannot be altered. Reference: `https://en.wikipedia.org/wiki/File:QR_Code_Structure_Example_2.svg`.

QR Code Module and Box Size. Each square block in a QR code is called a *module*. The number of pixels used to represent the width or height of a module is referred to as the *box size* (see Fig. 3).

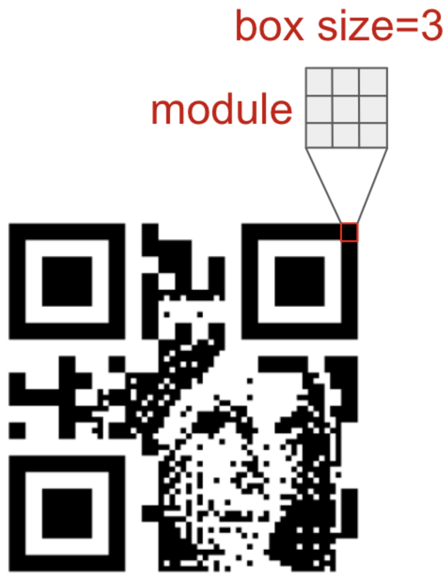

Fig. 3: QR Code Module and Box Size. A module is the basic unit of a QR code. The box size defines the number of pixels in the height and width of each module.

QR Code Version. QR codes are defined in 40 versions, ranging from 1 to 40. Each version corresponds to a different number of modules and structural patterns. The number of modules in both the width and height of the QR code can be calculated as:

$$\# \text{ of module} = 21 + (\text{version} - 1) \times 4. \tag{1}$$

QR Code Equivalence. Two QR codes are considered equivalent if the center value of each module (in the data region) is the same (see Fig. 5). This means that the surrounding pixels within each module can be modified without affecting the scannability of the QR code. These modifiable pixels provide room for embedding a styled image.

QR Code Error Correction. QR codes incorporate error correction to ensure data integrity even when parts of the code are damaged or altered. This is achieved using Reed-Solomon error correction [13], which allows a QR code to be correctly decoded even if a portion of its modules is unreadable. There are four error correction levels—L, M, Q, and H—corresponding to increasing levels of redundancy. This property is particularly beneficial for stylized QR codes, as it permits a certain degree of visual modification (e.g., embedding a style image) without compromising scannability, as long as the modifications remain within the correction capacity.

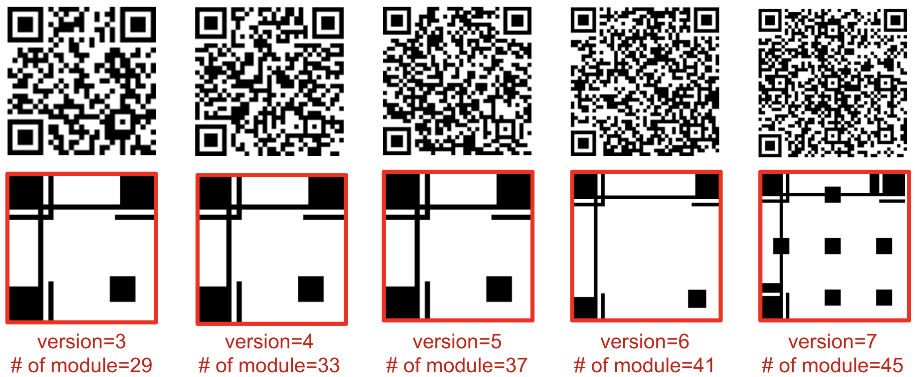

Fig. 4: QR Code Versions. Each QR code version has a different number of modules and unique structural patterns.

Fig. 5: QR Code Equivalence. The left and right QR codes are equivalent because the center value of each module is identical.

6.2 Method

Fig. 6 illustrates the framework of our algorithm using a QR code of version 6 and a box size of 5. First, a given text is encoded into a QR code, resulting in an image of size 205×205 pixels (41×41 modules). A style image is then resized to match this resolution, converted to grayscale, and processed using the error diffusion halftoning method of Jarvis, Judice, and Ninke [14].

For each module (a 5×5 pixel block), we copy the center pixel value from the original QR code and overwrite the corresponding 5×5 region in the halftoned image. If a module lies within an immutable region of the QR code (e.g., used for position, alignment, or timing patterns), we copy the entire 5×5 QR code pattern directly to preserve its structure. By assembling all modified regions, we obtain the final stylized QR code.

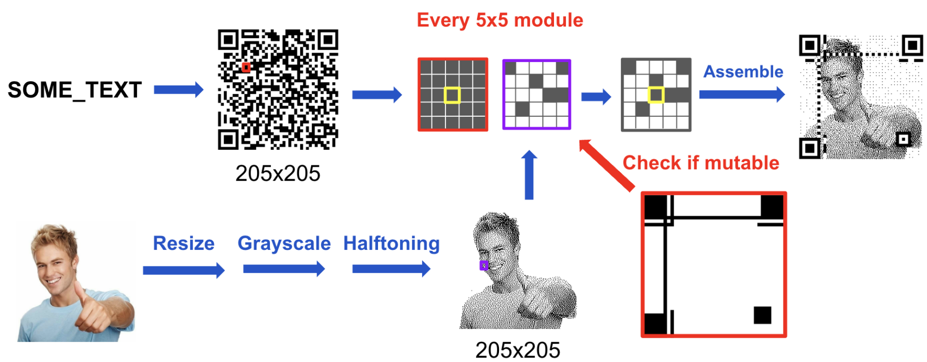

Fig. 6: Our Proposed Framework. Given a text string, we generate a QR code and align it with a halftoned version of a style image. The center pixels of each QR module are preserved to ensure readability, while the surrounding pixels are replaced with stylized content. Immutable QR regions are retained entirely.

QR Code Dropping with Salient Region Detection. Thanks to QR codes' error correction capability, we can discard certain parts of the QR pattern without affecting its scannability. To enhance the visual fidelity of the stylized image, we identify salient regions in the style image where QR patterns can be selectively dropped. We apply a simple edge detection algorithm using the Sobel operator to identify important visual regions (see Fig. 7). These regions are prioritized when choosing which parts of the QR code to discard during stylization.

Color Extension. We extend our approach to support color images by applying halftoning independently to the three color channels—Red, Green, and Blue. For each QR code module, we assign a color to the center pixel based on the original QR code value: white modules are set to RGB $(255, 255, 255)$ and black modules to RGB $(0, 0, 0)$. The surrounding pixels are filled with corresponding halftoned color values from the style image (see Fig. 8).

Fig. 7: QR Code Dropping. By leveraging the error correction property of QR codes, we can selectively remove some modules. We detect salient regions of the style image using Sobel edge detection and prioritize preserving those regions when stylizing the QR code.

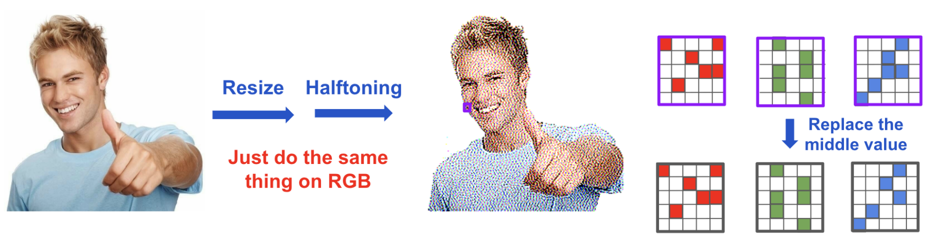

Fig. 8: Color Extension. We apply halftoning to each RGB channel of the style image and preserve the center pixel value of each module according to the QR code's original binary value. This enables the creation of visually appealing, scannable color QR codes.

7 Experiments

7.1 Datasets

To evaluate our proposed method, we collected a set of 10 images, 5 in color and 5 in grayscale, and 10 text samples of varying lengths (see Figs. 9 and 10). QR codes were generated using different versions and module box sizes for each combination. We then tested the readability of the resulting stylized QR codes using three different mobile phones: an iPhone 12 Pro Max using the built-in Camera app, a Samsung Galaxy S22 using the LINE QR scanner, and an iPhone 11 using the built-in Camera app.

Fig. 9: Images Used in the Experiment. Five color images (top row) and five grayscale images (bottom row) used to evaluate the visual stylization and scannability of the proposed method.

ID	Text
1	https://www.csie.ntu.edu.tw/
2	https://reurl.cc/Gnl3RZ
3	https://www.facebook.com/panchingping
4	https://www.youtube.com/watch?v=XGxIE1hr0w4
5	https://www.facebook.com/darkalexbu/

ID	Text
6	https://check007.net/Tag/Check/c67a6602-f5f0-4e1f-8de2-578b1151fbeb
7	https://www.youtube.com/watch?v=GBIIQ0kP15E
8	https://kaminyou.com/
9	https://link.springer.com/chapter/10.1007/978-3-031-72995-9_18
10	TSMC_TO_THE_MOON::TSMC_TO_THE_MOON

Fig. 10: Text Samples. Ten text strings of varying lengths used as the input content for QR code generation. These variations help test the method across different QR code versions.

7.2 Metrics

We evaluated our method across three dimensions: robustness, visual style quality, and QR pattern visibility.

To assess robustness, we tested whether each QR code could be successfully scanned by different mobile phones. We recorded the scanning performance using a 3-point scale: a score of 1 indicates that the QR code is immediately scannable, 0.5 means it is scannable but with delay or multiple attempts, and 0 means it cannot be scanned.

For visual quality with respect to style preservation, human evaluators were asked to rate how well the style image is preserved in the QR code. The score ranges from 1 to 5, where 5 indicates excellent visual quality and clear style representation, while 1 indicates poor stylization or distortion.

Finally, to evaluate the visibility of QR code patterns in the stylized image, we asked human raters to judge whether the QR code pattern is still visually obvious. A binary score is assigned: 1 if the QR pattern is not visually apparent, and 0 if the QR pattern remains clearly visible.

8 Results

8.1 Scan success rate

Scan success rate testing was conducted using the mobile phones of the three team members as the testing environment. The success rate was first evaluated using version 6 of the short length text. Overall performance was satisfactory. However, older iPhones, possibly due to their system versions, exhibited errors with larger box sizes, whereas Android devices showed some failure cases with smaller box sizes (see Figs. 11 and 12).

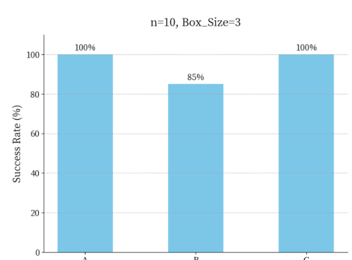

Fig. 11: Scan success rate. Text ID 2 (short length, Version 6) + 10 images (gray & color) + Box=3.

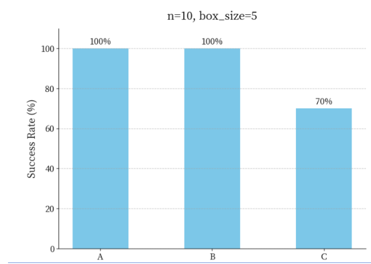

Fig. 12: Scan success rate. Text ID 2 (short length, Version 6) + 10 images (gray & color) + Box=5.

Next, the success rate was tested using version 8 of the long length text. The results were basically all successful. Higher versions may potentially improve the scan success rate (see Figs. 13 and 14).

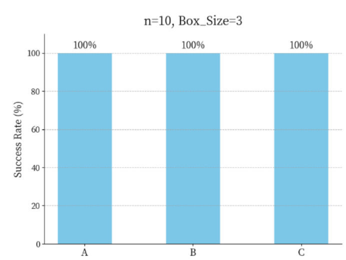

Fig. 13: Scan success rate. Text ID 6 (long length, Version 8) + 10 images (gray & color) + Box=3.

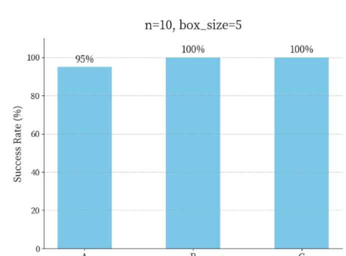

Fig. 14: Scan success rate. Text ID 6 (long length, Version 8) + 10 images (gray & color) + Box=5.

8.2 Quality

In the quality testing phase, the samples were evaluated based on the average visual assessment of the three team members. The first test examined whether different IDs would affect the quality. We first selected an image of Hokkaido for this test (see Fig. 15), with the gray box size set to 3 and the color box size set to 5, the results showed that there was almost no impact (see Fig. 16).

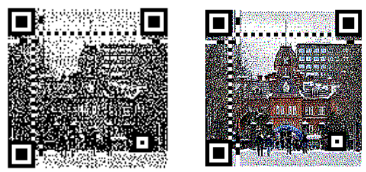

Fig. 15: Test Image. Hokkaido

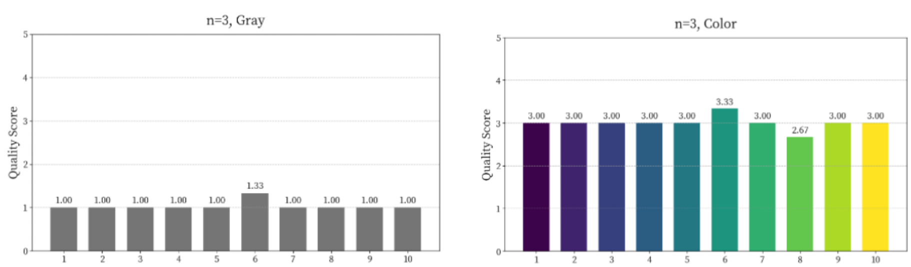

Fig. 16: Different ID Quality Test. Text ID 2 (short length, Version 6) + 10 images (gray & color).

Next, under version 6, the larger box size demonstrated a clear improvement in overall quality due to its ability to display more details (see Fig. 17). It was also observed that when the input image had a clearer and simpler composition with good contrast (see Fig. 18), the visual effect was significantly better.

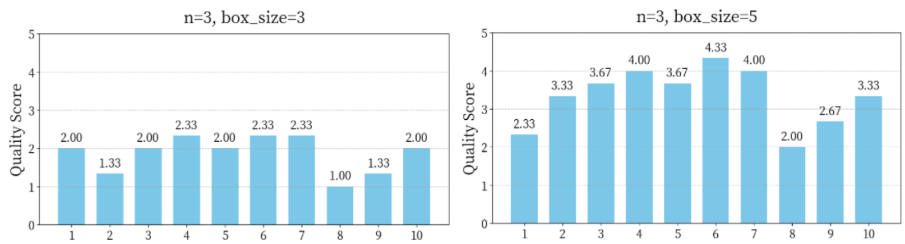

Fig. 17: Different Box Quality Test. Text ID 2 (short length, Version 6) + 10 images (gray & color).

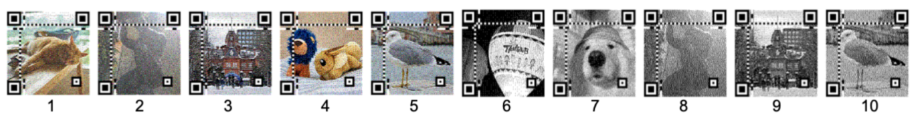

Fig. 18: Different Input Image. ID1 to 10.

When increasing to higher versions, the quality improvement for the smaller box size was not very noticeable, whereas the larger box size showed more substantial enhancement by revealing more details. For box size 3, even when upgraded to version 8, there was a slight degradation in quality, while box size 5 exhibited noticeably better overall edge and contour details (see Fig. 19).

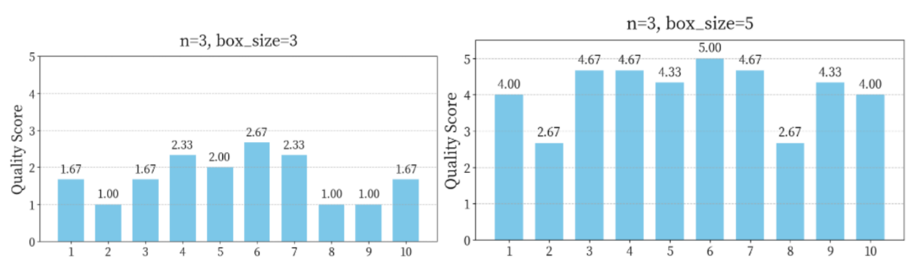

Fig. 19: Different Box Quality Test. Text ID 6 (long length, Version 8) + 10 images (gray & color).

Additionally, if the original image contains more complex compositional elements, there is also a noticeable improvement, such as in ID 1 (the orange cat) and ID 3 and 9 (buildings). However, images with poor contrast, such as ID 2 and 8, showed little to no improvement.

8.3 Pattern Visible

The original QR pattern is typically observed in large blank areas such as the sky or floor, as seen in IDs 3 and 6. Apart from these, it is generally almost invisible (see Fig. 20).

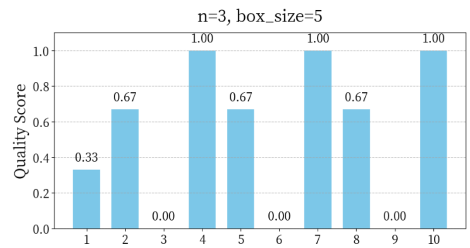

Fig. 20: Pattern Visible Quality Test. 1=invisible, 0=visible

9 Ablation Study

To evaluate the effectiveness of each component in our system, we conducted an ablation study focusing on the key design choices: halftoning, QR code version, edge-aware dropping, and box size. For edge-aware dropping and box size here, we also take the effect of halftoning into account.

9.1 Halftoning

We first examine the impact of halftoning. Without halftoning, the QR code pattern (the black and white dot) becomes clearly visible within the image, severely compromising the visual appeal, especially in areas with low image complexity or large blank regions, such as the sky. In contrast, applying halftoning significantly reduces the visibility of the QR code pattern, resulting in a more aesthetically pleasing and stylized output.(see Fig. 21 and Fig. 22)

9.2 QR Code Version

Next, we studied the effect of the QR code version by comparing different QR code versions on the same styled image with all the other parameters fixed (with halftoning, no edge-awared dropout, block size = 3). Higher versions (e.g., Version 8) provide more modules, enabling not only longer text encoding but also higher resolution for image rendering. This contributes to improved visual quality and robustness across different devices. (see Fig. 23)

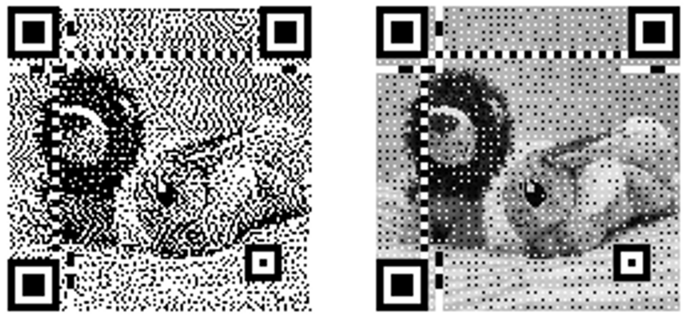

Fig. 21: The effect of halftoning in our gray scale styled QR code, where the left one has been applying halftoning and the right one is not applying halftoning. As the image shows, we can clearly see the QR code pattern on the right side.

Fig. 22: The effect of halftoning in our colored styled QR code, where the left one has been applying halftoning and the right one is not applying halftoning. As the image shows, we can clearly see the QR code pattern on the right side.

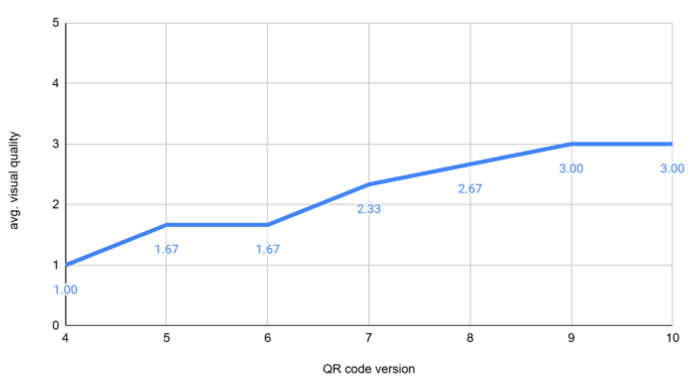

Fig. 23: QR code version compared to visual quality

9.3 Edge-Aware Dropping

The edge-aware dropping mechanism was also analyzed by setting different dropout ratios and all the other parameters fixed (with halftoning or without halftoning, block size = 3). By selectively skipping replacements in edge regions when a certain quota is reached, we can preserve more visual information in complex areas of the image. Although this technique improves image fidelity, it can negatively impact scanning success rates, particularly on devices with lower tolerance for visual noise and these styled QR code applying with halftoning. (see Fig. 24)

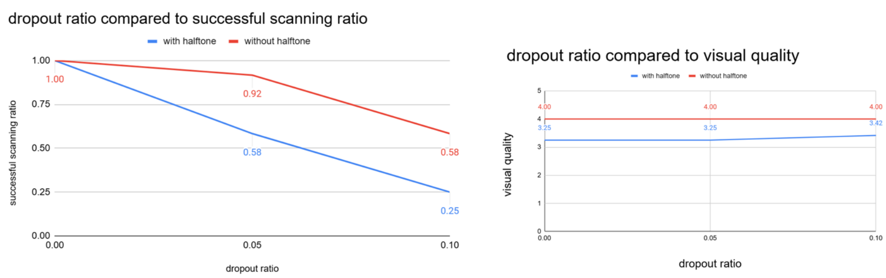

Fig. 24: Different dropout ratio test. The left side focus on successful scanning ratio and the right one focus on visual quality.

9.4 Box Size

Finally, we tested different box sizes. A larger box size (e.g., 5 pixels per module) improves image clarity and detail retention especially for these QR code with halftoning. However, this comes at the cost of a reduced successful scanning ratio for 20 30 percent for these QR code with halftoning and 10 20 percent for these without halftoning.(see Fig. 25)

9.5 Conclusion of Ablation Study

In summary, the ablation study highlights the delicate balance between artistic expression and technical reliability in generating artistic QR codes. Halftoning is essential for concealing the QR pattern and achieving high visual quality. Higher QR code versions improve resolution and robustness, making them suitable for complex images or long-text payloads. Edge-aware dropping smartly preserves visual features, but must be tuned to avoid compromising scan success. Larger box sizes offer better stylization, but reduce compatibility with older devices. Together, these findings emphasize the importance of adaptive parameter tuning based on deployment goals, whether the focus is on visual quality or successful scanning ratio.

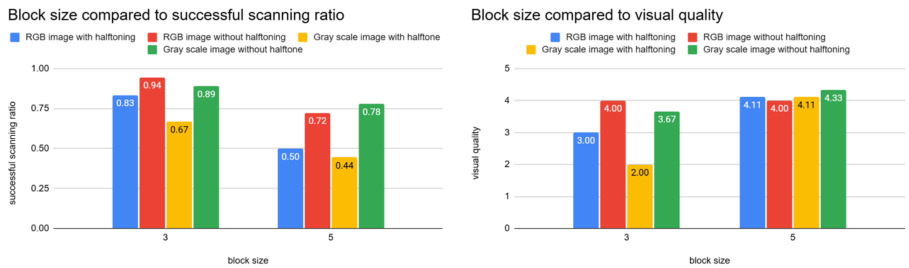

Fig. 25: Different box size test with halftoning effect. The left side focus on successful scanning ratio and the right one focus on visual quality.

10 Discussion and Conclusion

Through extensive experiments, we observed several factors that influence the effectiveness and scannability of stylized QR codes. Below, we summarize key findings and insights from our evaluation.

Higher QR code versions allow encoding longer text strings and generally lead to higher scanning success rates. This is because more modules are available for data encoding and error correction, providing greater flexibility for stylization.

Images with simple structures and high contrast produce better stylization results. Complex textures or cluttered regions tend to obscure the QR code structure, making it harder to balance aesthetics with functionality.

Without halftoning, stylized QR codes can achieve enhanced visual quality. However, the QR code patterns remain clearly visible, which may reduce the artistic appeal. In contrast, applying halftoning helps to better hide the QR code patterns, leading to a more seamless integration of the style image. Nonetheless, in regions with large blank spaces (e.g., sky), the QR code may still become noticeable.

We also experimented with edge enhancement using different dropout ratios. This technique improves the visual emphasis on important image features, but excessive QR pattern dropping negatively impacts scanning robustness.

Lastly, using larger box sizes contributes to better visual quality due to higher resolution in the styled image. However, it also reduces the scanning success rate, as the enlarged modules may introduce more visual noise relative to the preserved QR structure.

In conclusion, our method effectively balances stylization and readability of QR codes through modular halftoning and saliency-aware modifications. The trade-off between visual quality and scanning reliability remains a central design consideration, and our framework allows flexible adjustment depending on the intended application.

References

1. Hung-Kuo Chu, Chia-Sheng Chang, Ruen-Rone Lee, and Niloy J Mitra. Halftone qr codes. *ACM transactions on graphics (TOG)*, 32(6):1–8, 2013.
2. Kinjal H Pandya and Hiren J Galiyawala. A survey on qr codes: in context of research and application. *International Journal of Emerging Technology and Advanced Engineering*, 4(3):258–262, 2014.
3. Berrin Arzu Eren. Qr code m-payment from a customer experience perspective. *Journal of Financial Services Marketing*, page 1, 2022.
4. Nur Ain Syazlin Mohd Zakri, Nur Nazira Nawawi, Amanina Mat Ghani, Maisarah Abd Hamid, Wan Nazriah Wan Nawawi, and Sulaiha Mohd Isa. The role of qr code applications in enhancing customer satisfaction at restaurants: A conceptual study. *Asian Journal of Research in Education and Social Sciences*, 6(S1):385–392, 2024.
5. Michelle Kelly Schultz. A case study on the appropriateness of using quick response (qr) codes in libraries and museums. *Library & Information Science Research*, 35(3):207–215, 2013.
6. Mingliang Xu, Hao Su, Yafei Li, Xi Li, Jing Liao, Jianwei Niu, Pei Lv, and Bing Zhou. Stylized aesthetic qr code. *IEEE Transactions on Multimedia*, 21(8):1960–1970, 2019.
7. Li Hai-Sheng, Xue Fan, and Xia Hai-ying. Style transfer for qr code. *Multimedia Tools and Applications*, 79(45-46):33839–33852, 2020.
8. Hao Su, Jianwei Niu, Xuefeng Liu, Qingfeng Li, Ji Wan, and Mingliang Xu. Q-art code: Generating scanning-robust art-style qr codes by deformable convolution. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 722–730, 2021.
9. Guangyang Wu, Xiaohong Liu, Jun Jia, Xuehao Cui, and Guangtao Zhai. Text2qr: Harmonizing aesthetic customization and scanning robustness for text-guided qr code generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8456–8465, 2024.
10. Hai-Sheng Li, Fan Xue, and Hai-ying Xia. Style transfer for qr code. *Multimedia Tools and Applications*, 79(45):33839–33852, 2020.
11. Jia-Wei Liao, Winston Wang, Tzu-Sian Wang, Li-Xuan Peng, Ju-Hsuan Weng, Cheng-Fu Chou, and Jun-Cheng Chen. Diffqrcoder: diffusion-based aesthetic qr code generation with scanning robustness guided iterative refinement. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5916–5925. IEEE, 2025.
12. Jia-Wei Liao, Winston Wang, Tzu-Sian Wang, Li-Xuan Peng, Cheng-Fu Chou, and Jun-Cheng Chen. Diffusion-based aesthetic qr code generation via scanning-robust perceptual guidance. *arXiv preprint arXiv:2403.15878*, 2024.
13. Irving S Reed and Gustave Solomon. Polynomial codes over certain finite fields. *Journal of the society for industrial and applied mathematics*, 8(2):300–304, 1960.
14. Keith T Knox. Error image in error diffusion. In *Image Processing Algorithms and Techniques III*, volume 1657, pages 268–279. SPIE, 1992.