Distance-Dependent Barcodes for Context-Aware Mobile Applications

Roope Palomäki
Aalto University

Maria L. Montoya Freire
Aalto University

Mario Di Francesco
Aalto University

ABSTRACT

This article introduces the novel concept of *distance-dependent barcodes*, which provide users with different data based on their scanning distance. These barcodes employ color blending as the key technique to achieve distance-dependence. A simple yet robust encoding scheme is devised accordingly to distinguish between near and far users. Through several experimental results, the proposed technique is shown to be effective (in terms of clear separation between near and far scanners), reliable (as to successful scans), and practical (it can be used in off-the-shelf smartphones). A few representative use cases are then presented to establish distance-dependent barcodes as an enabling technology for context-aware mobile applications. They include casual interactions with public displays, where the role of users is determined based on their distance from a screen, and augmented reality in retail, where distance-dependent barcodes provide information on available goods with different granularities. Finally, distance-dependent barcodes are shown to be user-friendly and effective through a user study.

CCS CONCEPTS

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

KEYWORDS

barcode; context awareness; color blending; distance-dependent encoding; smartphone; mobile computing; public displays; augmented reality

ACM Reference Format:
Roope Palomäki, Maria L. Montoya Freire, and Mario Di Francesco. 2020. Distance-Dependent Barcodes for Context-Aware Mobile Applications. In *22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '20), October 5–8, 2020, Oldenburg, Germany.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3379503.3403534

1 INTRODUCTION

Barcodes have become so prevalent that they are used on a daily basis. Their main purpose is to encode a short amount of machine-readable data that can be used for different software-based applications: logistics is one of the most prominent [35]. Barcodes are also widely used by mobile phone users as they conveniently allow to look-up resources by simply taking a picture, instead of manual typing. Most importantly, they can turn traditionally cumbersome interactions into a seamless and intuitive user experience [13]. As a result, they have been employed in a wide range of mobile applications for different purposes, including mobile payments [30], user authentication [28], embedding metadata into video streams [37], and as markers in augmented reality [29].

This article introduces the concept of *distance-dependent barcodes* – color barcodes that not only encode machine-friendly data, but also bear different information based on the distance they are scanned from. Figure 1 presents an illustrative example. A display shows a special – distance-dependent – barcode to two users holding smartphones: Nancy, who is close to the display; and Frank, who is farther away. Decoding the barcode produces certain data when scanned by Nancy; other data when scanned by Frank. The key enabler of distance-dependent barcodes is *color blending*, which occurs when two colors overlap at either the source or the captured image, forming a perception of a third color [31]. Specifically, the proposed technique leverages color blending on a module level. In particular, it arranges different color blocks within a module into patterns that cannot be distinguished at distance. The main intuition is that blending occurs when the blocks in the patterns are captured with a low-enough resolution. In other words, given a fixed capture resolution, color blending between fixed-size barcode modules increases with distance.

Employing color barcodes in practice entails several challenges, primarily related to the unreliable color characterization by cameras in off-the-shelf mobile devices [34, 38]. These issues are determined by the variability of environmental conditions, the camera settings (including ISO sensitivity and white balance), the physical characteristics of the sensor and lenses, as well as the heavy post-processing carried out by hardware manufacturers to make the pictures more appealing to the users [4]. In this respect, robust characterization of color blending is even more challenging, as the phenomenon is complex and intrinsically determined by environmental factors that cannot be directly controlled.

This work specifically addresses these challenges and establishes the following contributions.

- It introduces the novel concept of distance-dependent barcodes – the first of its kind, to the best of the authors' knowledge. The proposed technique extends regular barcodes by adding a new dimension – namely, *distance* – to support context-aware mobile applications. Such context awareness is transparently accomplished through the familiar and user-friendly process of scanning a barcode (Section 2).
- Selected use cases illustrate a meaningful application of distance-dependent barcodes to mobile computing scenarios (Section 3), including casual interactions with public displays [15, 19] and augmented reality for retail [3]. A user

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).
MobileHCI '20, October 5–8, 2020, Oldenburg, Germany
© 2020 Copyright held by the owner/author(s).
ACM ISBN 978-1-4503-7516-0/20/10.
https://doi.org/10.1145/3379503.3403534

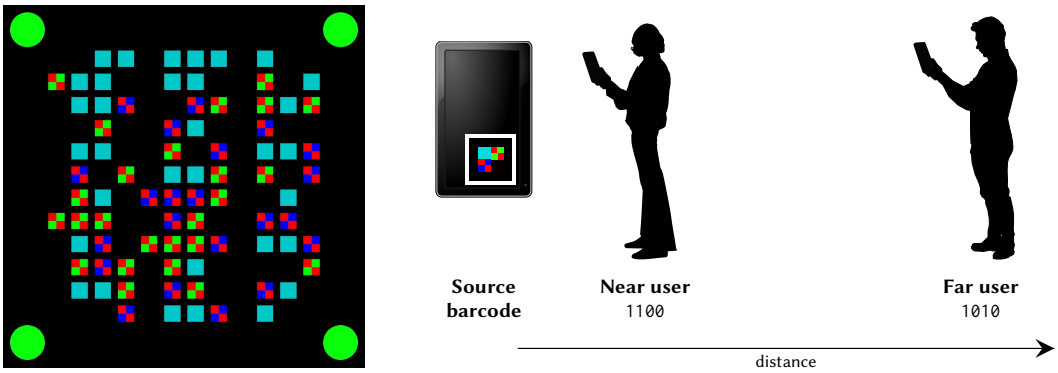


Figure 1: (a) A sample distance-dependent barcode representing two distinct bit sequences at the same time; (b) when scanned, a near user decodes one, a far user another due to color blending.

study conducted on the former demonstrates that distance-dependent barcodes are practical and user-friendly (Section 6).
- A simple yet robust distance-dependent design is created, grounded on an empirical study of color blending on diverse media (i.e., paper and different types of displays). The corresponding barcodes enable encoding two bitstreams into modules providing data that is different for users at near and far distance (Section 4).
- Distance-dependent barcodes are shown to be effective (in terms of clear separation between near and far scanners), reliable (as to successful scans), and practical (they can be used in off-the-shelf smartphones) through several experimental results. A comparison with the state of the art reveals that the proposed technique is more beneficial than using multiple QR codes and FOCUS, a well-known frequency-based barcode (Section 5).

2 RELATED WORK

Several 2D barcodes have been proposed and evaluated for mobile phone applications [12, 39]. Color barcodes have also been developed to increase the encoding capacity [23, 25], particularly, in the context of streaming data through screen-camera communication [7, 17, 40]. Several solutions trade-off capacity for reliability, which is critical in unidirectional screen-camera links [33, 34]. Unlike existing solutions in such a context, this work does not leverage color barcodes to increase the capacity of screen-camera communication; instead, it realizes a barcode that encodes distance-dependent data to enable novel context-aware mobile applications.

A few 2D barcode designs support embedding different data streams into a single barcode, primarily to support users with different capture rates and channel conditions in unidirectional screen-camera communications. Among them, Strata [10] introduced a layered coding scheme inspired by hierarchical modulation to provide two data streams: a base layer for receivers with a low signal-to-noise ratio (SNR) and an enhancement layer for those with a higher SNR. Similarly, distance-dependent barcodes embed two different bit streams into a single structure, which result in different data decoded at the mobile phone based on its distance. In contrast, the solution here leverages color modules distributed over the whole barcode, while Strata uses black-and-white blocks and separate regions as layers.

There are also frequency-based barcode designs that encode multiple data streams into a single barcode [9]. Among them, PixNet [21] and FOCUS [8] are the most prominent examples. They both employ orthogonal frequency division multiplexing: source data are represented as a square 0-1 matrix, which is encoded into a grayscale image through the 2D inverse fast Fourier transform (FFT). In doing so, they divide the code spectrum into multiple subchannels, each corresponding to different frequency ranges: a distant receiver can obtain only the channels with the lowest frequencies, while a close receiver also those with higher frequencies. Original data are decoded at the receiver by applying the FFT to the captured image. The design proposed in this article does not operate on the frequency domain, but rather embeds different bit streams through the spatial features associated with colored blocks. Such an approach is simpler yet more effective than frequency-based solutions, including FOCUS [8] (see Section 5.2 for a comparison).

This work shares similarities with some solutions developed in the context of computer vision and artificial image generation. For instance, hybrid images [20] appear different depending on the viewing distance: a close viewer sees one while a far user another. The process of overlaying the two images consists in extracting the high-frequency components of the close image and the low-frequency components of the far image. The resulting image is then created by blending the two. The result achieved by hybrid images is comparable to what is proposed here in terms of distance awareness. However, hybrid images leverage *visual perception* of humans to provide the illusion of a static image that changes at distance. In contrast, this work develops a *machine-friendly* scheme that encodes distance-dependent data in such a way that they can be reliably decoded by near and far scanners.

Somewhat related research has focused on embedding images into machine-readable codes, for instance, QR codes [6]. Some approaches overlay a small picture (e.g., a thumbnail) on the code or rearrange the blocks therein to resemble a source image, while relying on error correction for successful decoding [27]. Others

(a) (b) (c)

Figure 2: (a) Distance-dependent barcodes for casual interactions with public displays; user interface for (b) presenter and (c) audience.

embed QR codes into color images [6], or modify a source picture by adding custom structures of machine-readable data [11, 36]. These works mediate human perception and machine-friendly data representation. However, they only support a single bit stream through a machine-readable code that does not disrupt the visual appearance of a source image. In contrast, this work encodes two different bit streams into one machine-friendly distance-dependent barcode.

The closest solution to this work is given by [1], which embeds unnoticeable markers into images shown on displays. To do so, it relies on chromatic flicker, by changing a color at high speed while keeping the original luminance. Such a scheme is extended in [18] for the case of near-screen mobile interaction in augmented reality. Despite leveraging color blending, the works above operate on the time domain by changing the colors of a static image that is shown on a screen. Moreover, they are only suitable for interacting close to a display. In contrast, this work operates on the spatial domain and leverages color blending to design a barcode that is distance-dependent and can also be printed on paper.

3 APPLICATIONS

Distance-dependent barcodes enable different types of applications where distance-sensitive information is beneficial, including (but not limited to) advertising, collaborative work, indoor wayfinding, and room access control. The following details two representative use cases: casual interactions with public displays and augmented reality for retail.

3.1 Use case 1: casual interactions with public displays

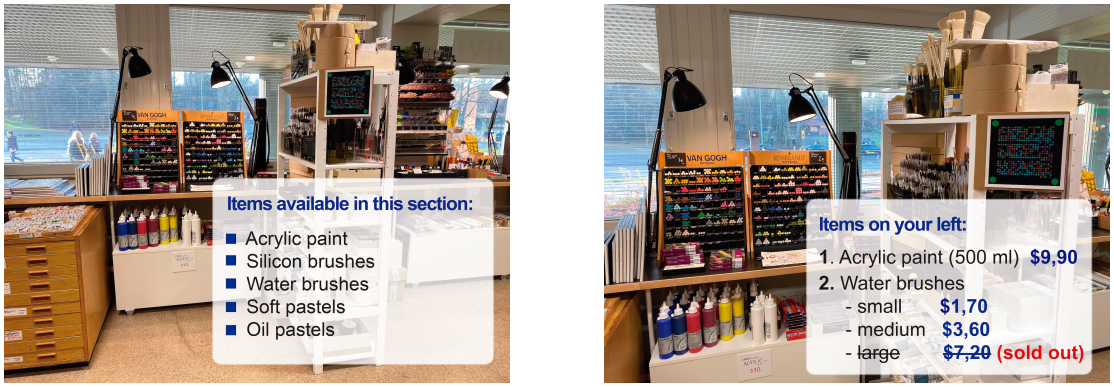
The first use case covers casual interactions with public displays (Figure 2). The reference scenario includes a public display that is available for use by people nearby [15]. Specifically, users can take control of the display through their mobile phone and use the latter to show different type of media, including videos and presentation slides. This is particularly useful in collaborative work, for instance, a team discussing a project in a meeting room. It can as well support people who just happen to start a casual conversation next to a display during their coffee break.

A distance-dependent barcode shown on the display allows users to establish a virtual link with the content therein (Figure 2a). Such a barcode also provides a quick and user-friendly option to distinguish between users' roles. The user close to the display is its temporary *owner*: he (she) can select the content shown, control it, and access additional tools such as speaker notes as well as real-time audience feedback from their smartphone (Figure 2b). Users farther from the display are the *audience*: they can obtain a local copy of the displayed content as well as auxiliary resources; they can also take notes and discuss with other participants to the meeting, including the owner (Figure 2c).

Establishing a link between a personal device and a display (e.g., a screen or TV) can actually be done with existing barcodes, for instance, QR codes [19]. However, selecting specific functions or roles may require additional input from the user who would need, for instance, to log into a service or explicitly pick options from some user interface. Multiple barcodes could also be used, each associated with a different function or role. However, users would need to figure out which barcode to scan depending on the intended outcome, which requires additional effort and is not very practical. In contrast, distance-dependent barcodes are transparent and support the different roles of owner and audience as naturally defined by how close people are to the display. This way, no other systems are needed to infer user location either.

3.2 Use case 2: augmented reality for retail

The second use case is represented by an augmented reality application for retail (Figure 3). Users can point their smartphone to the different sections in a store to get information about the goods therein [22]. Distance-dependent barcodes are leveraged here as markers that provide information with different granularity. At a far distance, they broadly describe the surroundings of the scanner, for instance, as the kind of goods in the aisles or shelves nearby (Figure 3a). When the user gets closer, the barcode provides more specific information on selected items, for instance, the price of those on sale as well as their availability (Figure 3b). For the user, enjoying distance-dependent barcodes is as simple as ensuring that the barcode is within the camera view. The overlayed information seamlessly updates based on distance as the user moves closer to



(a) (b)

Figure 3: Distance-dependent barcodes for augmented reality in retail: the user view in the far range (a) shows a broad description of the area where goods are located, while that in the near range (b) provides a list of the specific products close to the user.

or farther from the barcode. For the retailer, it is enough to print the barcode, attach it to some structure, and associate it with the inventory system.

The proposed solution has some advantages over alternative options for distance tracking, especially technologies for indoor positioning. In fact, it requires no additional investment in hardware, unlike radio beacons or other special-purpose location tracking systems. Deploying distance aware barcodes does not rely on a prior characterization of the environment, as opposed to both WiFi fingerprinting and computer vision methods for continuous location tracking [14, 22]. Finally, near and far scanners can be distinguished with an accuracy comparable to that of off-the-shelf hardware (as it will be shown in Section 5).

4 DISTANCE-DEPENDENT BARCODES

The goal of this article is to design a barcode that bears different information depending on the distance it is scanned from. The following starts by introducing the main principles behind distance-dependent encoding based on color blending, supported by experimental data considering different types of media. The rest of the section introduces a simple yet effective module design, followed by the implementation details of a barcode that enables distance-dependence in practice.

4.1 Basic principle and empirical study

A distance-dependent barcode[1] consists of a bi-dimensional arrangement of *modules*, namely, square colored blocks with fixed size, each encoding a certain number of bits. There are two different types of modules in a distance-dependent barcode: macro blocks and patterns. *Macro blocks* simply consist of a single color covering the whole area of a module. These modules are recognized the same at close and far distances. *Patterns* instead are an arrangement of smaller sub-blocks with different colors. These modules appear the same at the source and when scanned close, as the receiver is able to clearly distinguish the colors in individual sub-blocks. At distance, their colors *blend* into a third color seemingly covering a large share of the module [31]. For instance, a macro block could be black, while a pattern could consist of alternating red and green sub-blocks the size of which is half of the module size, as illustrated in Figure 1a. The presence of both macro blocks and patterns enables encoding two arbitrary bit sequences into a distance-dependent barcode. This allows users at near and far distances to decode different data when scanning the same barcode, as illustrated in Figure 1b.

Color blending is an intuitive phenomenon in terms of human perception, but it can be explained on similar grounds in the context of computer vision [31]. A challenge here is how blending can be employed in a way that is machine-friendly and robust to the constraints due to mobile hardware, as well as the environmental factors occurring in real settings. To address this issue, an empirical study was carried out as a preliminary step to design modules that could achieve distance-dependence in practice. In this respect, different media were considered and a qualitative evaluation of the blending outcomes was conducted to establish the feasibility of the approach in diverse conditions. The first medium is *paper*, for printed barcodes that could simply be embedded into documents or posters, or otherwise attached to physical objects. The other medium is a *display* (e.g., a screen) that could show the barcode as a means to connect to and (or) interact with a service, such as in the use case presented in Section 3.1. For the latter, it is also important to consider the impact of different display manufacturing technologies. As a consequence, different types of displays were employed to represent the most widely available panel types: in-plane switching (IPS), active matrix organic light emitting diodes (AMOLED), and variable aperture (VA). For simplicity, black and cyan macro blocks were used in the study, as well as red-green and red-blue patterns. Pictures of the different module types were then taken with a OnePlus 6 smartphone[2]. Experiments were conducted in a meeting room with both natural and artificial light with a light intensity level varying between 450 and 880 lux.

[1] While pre-existing barcode structures could be used, the following considers a custom scheme as this better allows to explore the design space.

[2] Particularly, the rear-facing camera of the smartphone under the same settings as those detailed in Section 5.1.

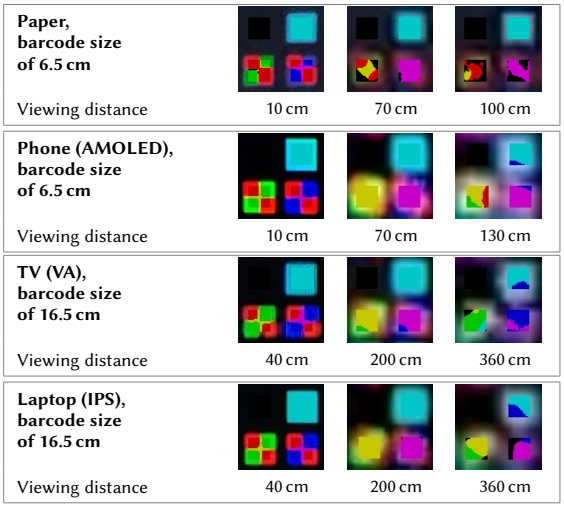


Figure 4: Examples of scanned modules for different media as a function of distance.

Figure 4 shows samples of modules scanned at distance for different media to show color as observed at the smartphone. In the figure, each module corresponds to a size of 25 by 25 pixels in the captured image, irrespective of the physical barcode size. The region at the center of the module shows colors quantized to the possible options as defined in the source modules. Overall, there is very limited blending between the sub-blocks, irrespective of the medium and display type, for the pictures taken at a close distance. Some blending always occurs when pixels along the block edges do not perfectly align between the barcode and the smartphone, when the pixel densities are different, and due to compression of the captured image. This manifests, for instance, as the thin yellow and magenta lines in the close patterns. On the other hand, the images taken from a farther distance show a large amount of blending within the center of the module, with some dependence on the specific medium.

The samples taken on paper and on a VA display (in the first row of the figure) show quite some amount of the colors in the pattern at the corners. However, the blending at the center of the module is still clear. In contrast, blending affects almost the whole module in the samples taken on AMOLED and IPS displays (in the second row of the figure), and only a small amount of the individual colors in the patterns can be seen at the corners. Finally, the images taken at a very large distance show that blending is not the same for all considered medium types. Artifacts include the presence of primary colors, irregular blending patterns, and even some black pixels. These artifacts are partially due to the blending between adjacent modules that occurs at very far distances. Nevertheless, there is still a prominent color at the center of the block.

In conclusion, the empirical evaluation above demonstrates that color blending consistently manifests in real scenarios with commonly-available hardware. The rest of the section discusses how to leverage it to realize distance-dependent barcodes that work in practice.

4.2 Module design

The previous discussion has not addressed the selection of specific colors, but rather considered them as given for the different modules. But what is a sensible choice of colors in the considered scenario? First of all, the actual number of required colors should be derived [38], which depends on the amount of information encoded in a module. For simplicity, the following considers modules encoding a single bit, resulting in a symmetric scheme – the information contained in a barcode is equally divided into near and far data. The 1-bit encoding requires at least five source colors: two for the macro blocks, and at least three more for the patterns, assuming that they are made of four sub-blocks each. The two blendings of the patterns should be considered as additional colors in this context. In fact, distance-dependence would not occur in practice if blended colors could not be distinguished from the others. As a consequence, the 1-bit encoding needs no less than seven colors in total.

Now that the number of colors is known, their actual values remain to be decided [34]. A very simple idea is to pick the primary colors in the RGB color model (i.e., red, green, and blue) for the sub-blocks in the patterns. One of the two colors could be reused in the patterns without affecting ability for the receiver to recognize the modules. Choosing red as such a color results in the red-blue and the red-green patterns, respectively blending into yellow and magenta, as in the previously-discussed empirical study. As there is no blue-green pattern, cyan is not produced by any blendings and can be employed for a macro block. These colors correspond to six equally-spaced points in the hue circle [26], with angular distances of 60° between adjacent points (Figure 5a).

One more color still needs to be chosen for the second macro block. Picking an additional point in the hue circle is not a viable solution, as this would reduce the separation between the chosen colors. However, there is a different option: brightness could be used instead of hue. This is equivalent to consider a different dimension, which can be easily seen through the HSL double cone [26], which expresses the lightness on an axis perpendicular to the hue circle (Figure 5b). Here, the points offering maximum separation are those at the vertices of the cone: white and black. Between them, black is preferable as white emitted by displays could bleed into adjacent colors, making them appear lighter.

Figure 5c shows a module design that follows the considerations above. In particular, the illustration denotes a condensed truth table: the column heading indicates the target bit at far distance, the row heading the target bit at near distance, and inner cells the modules corresponding to these bit combinations. The modules in the figure are shown at the *source*, and are also recognized as such by *near* scanners. In contrast, Figure 5d shows the source modules as they appear to a *far* scanner. Note that the macro blocks entirely consisting of black and cyan are the same as those in Figure 5c: they encode the same bit for near and far data, thus, their appearance does not change over distance. In contrast, the red-blue and red-green patterns appear as magenta and yellow macro blocks, respectively.

Consider again the illustrative example in Figure 1. Given the same source barcode with four modules, near user Nancy decodes the bit string 1100, while far user Frank decodes the bit string 1010. The way the two strings overlap bit-wise determines which

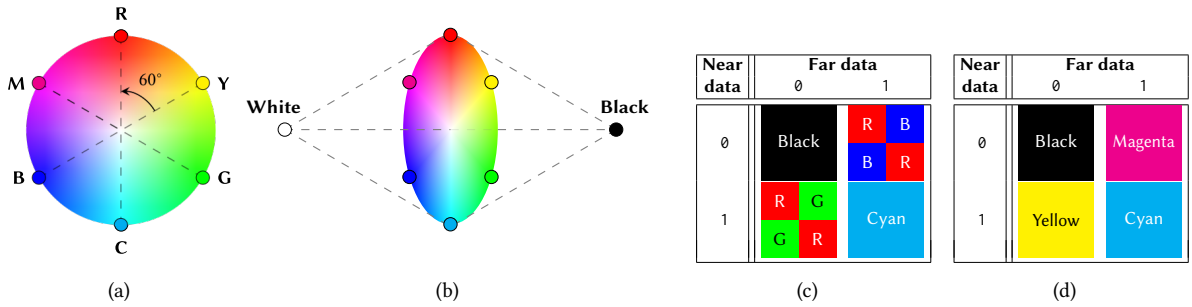


Figure 5: (a) Primary colors in the RGB additive model and their blendings in the hue circle; (b) the corresponding projection on the HSL cone. (c) Module design for the symmetric 1-bit encoding, representing all possible combinations of near and far data. (d) Corresponding modules observed at far distance as a result of color blending in the red-blue and red-green patterns.

specific blocks are used in the corresponding modules, according to Figure 5c.

4.3 Implementation

The following describes an implementation of a distance-dependent barcode according to the design in the previous section. The related software as well as instructions on how to use it are available online at `http://dawcode.aalto.fi`.

Barcode generation. Creating a barcode starts by arranging data into messages consisting of a header, a payload, and a trailer [16]. The header contains the payload length, which is necessary to understand where meaningful data end, since the number of modules in a barcode is fixed. The payload contains the actual near / far data provided by the application. The trailer contains redundant data obtained by applying forward error correction, namely, through a Reed-Solomon code [24], which is also used in QR codes.

The resulting near / far messages are then considered bit-wise and individual modules are encoded depending on the specific near-far bit combinations, according to the possible options in Figure 5c. The modules are then arranged into a barcode structure. As most bi-dimensional barcodes [12], the proposed distance-dependent barcodes are square; namely, they have the same number of horizontal and vertical modules. To counter cross-module color blending, black padding is added between the modules. The padding extends to the outer part of the barcode, containing green circles as corner locators – similar to [8]. These locators allow the scanner to distinguish the barcode from the background and also enable correction of perspective distortion [2, 7].

Figure 1a shows a distance-dependent barcode consisting of 12 by 12 modules separated by 15% padding, with the 1-bit encoding described earlier.

Barcode scan. Code scan follows the standard pipeline for processing 2D color barcodes [7]. First, corners are detected in the captured image by applying a threshold for the specific green color, slightly eroding the resulting image, searching for contours of a sufficiently large area, and choosing four contours as the corners based on minimal and maximal vertical as well as horizontal coordinates. Then, a perspective transformation is applied to obtain a square image – with a size proportional to the capture resolution – which is smoothed with a small Gaussian blur to reduce the moiré effect and sensor noise [2]. Next, each module is read within the area defined by the midpoint and its neighbors by accounting for the padding. Each pixel is quantized to a hue in the HSV color space, or to black if its luminosity is sufficiently low. Decoding of individual modules is then performed based on the quantized colors in the corresponding blocks. The main idea is to use the *proportion* of colors in the quantized block to decide on the data. In particular, the concepts of *majority*, *minority* and *existence* are defined based on different thresholds: 75%, 35%, and 5%, respectively. The cases where the module appears as a single color – actual single-color modules as well as blended two-color modules – are checked first, allowing for the fact that the luminosity of bright modules may bleed into black modules [10]. Accordingly, a module is interpreted as black if black pixels are more than the minority threshold; whereas cyan, yellow, and magenta modules are recognized when the corresponding colors are beyond the majority threshold. Non-blended patterns (with two colors) are considered next: they are recognized when their two colors combined are over the minority threshold and each individual color is over the existence threshold. If these checks are not successful, the decision is based on the color occurring with the highest frequency, with some simple rules to account for inter-module blending at longer distances.

5 EVALUATION

This section provides quantitative results to demonstrate the effectiveness of distance-dependent barcodes. The evaluation is conducted by considering both printed barcodes and those shown on a display.

5.1 Experimental setup and methodology

The following introduces first the testbed setup, then the used methodology as well as the considered metrics. The section concludes with a description of relevant solutions in the state of the art that are considered for comparison purposes.

Testbed setup. The testbed setup for the evaluation consists of a 65" Philips 6520 Series TV with a 4K VA panel, a 13" MacBook Pro (2017 model) with an IPS retina display, and a OnePlus 6 smartphone[3] in a meeting room with both natural and artificial light, with an average light intensity level of 670.9 lux and a standard deviation of

[3] `https://www.oneplus.com/6/specs`

93.2 lux. Printed barcodes are produced with a color laser printer for office use and standard-quality paper. The displays have the default color calibration settings and the maximum display brightness. Images captured by the scanner are taken at a 16 MP resolution (a commonly-available option for recent personal devices) with fixed aperture at f/1.7, shutter speed at 1/60 s, ISO at either 300 (for paper) or 160 (for displays), and white balance set to fluorescent (for paper) or daylight (for displays). These settings were found suitable for scanning barcodes from displays in indoors scenarios through a preliminary study aimed at increasing the consistency and reliability of the colors observed at the receiver, similar to the approach in [38].

The smartphone runs Android version 9 (corresponding to API level 28), with OpenCV SDK version 4.0.0 pre-installed. The distance-dependent barcode has 24 by 24 modules, each representing 1 bit of information. This configuration encodes a total of 576 bits or 72 bytes of binary data. Reed-Solomon error correction is applied to source data by adding 16 bytes of redundant data. Given that the frame includes a 1-byte header (Section 4.3), the actual payload is 55 bytes, which is large enough to support the applications described in Section 3. The content of near and far data in the barcode is randomly generated.

Methodology and considered metrics. Experiments are carried out by scanning a barcode with the rear-facing camera of a hand-held smartphone. This accounts for some shaking that can occur in practice upon scanning. The smartphone is kept straight and perpendicular to the surface of the medium bearing the barcode for consistency. Each experiment is run ten times according to the independent replication method. Unless otherwise specified, the obtained standard deviations are below 5% of the average values; thus, they are not reported in the figures for the sake of readability.

The effectiveness of the distance-dependent barcode is evaluated in terms of separation and reliability. *Separation* is defined as the range of distance at which only near or far data are decoded from the barcode. In other words, the shorter the range where the same barcode could be decoded as neither near nor far, the better the separation. More specifically, separation is measured as the correlation between the decoded bytes and the source data. *Reliability* is defined as the number of frames correctly decoded – namely, those which could be located in the captured image (following corner detection) and whose errors (if any) could be successfully corrected – over the total number of barcodes scanned.

Solutions in the state of the art. The following schemes are considered for comparison purposes.

- *QR code-based.* This scheme uses two distinct QR codes next to each other as a baseline. The two QR codes are setup in such a way that one can be decoded from near and the other from far distance. This is accomplished by using different physical sizes and error correction levels: 9 by 9 cm and error correction set to medium for the near QR code; 16.5 by 16.5 cm and error correction set to high for the far QR code. The data size is set to 55 bytes in both cases.
- *FOCUS-based.* This scheme employs FOCUS codes [8], a state-of-the-art solution leveraging a frequency-based encoding for reliable screen-camera communication. FOCUS divides encoded data into different channels, corresponding to different ranges in their frequency spectrum, set to 16 in the evaluation. The FOCUS-based scheme employs the reference open-source code provided with [8] to produce a 16.5 by 16.5 cm barcode. The frame length is 80 bytes, including a 4-byte header and 16 bytes of Reed-Solomon error correction code.

5.2 Experimental results

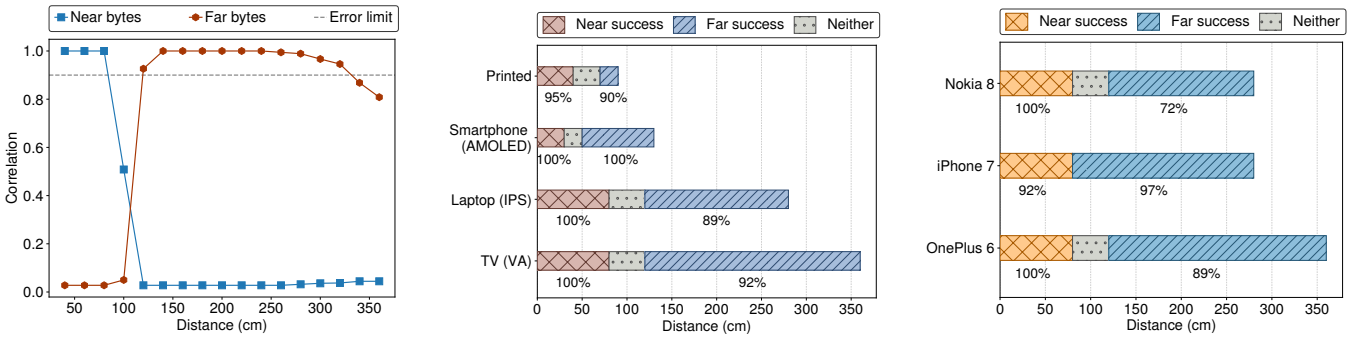
The following details the results from the experimental evaluation of the proposed distance-dependent barcodes. Two different barcode sizes are considered: 6.5 cm for the printed barcode as well as the one shown on the AMOLED display; and 16.5 cm for those shown on VA and IPS displays. The distance between the medium and the scanner is varied with a step size of 10 cm for the smaller barcodes and 20 cm for the larger ones. When not explicitly mentioned, the 65" Philips TV and a 16.5 by 16.5 cm barcode are used in the experiments.

Separation and reliability. The first set of experiments aims at evaluating the performance of distance-dependent barcodes in terms of both separation and reliability for different external conditions (Figure 6).

Figure 6a shows the correlation between decoded and source data for the raw bytes in the payload (i.e., without error correction) as a function of distance. The figure also shows the threshold after which error correction allowed to successfully decode the modules in the barcode as a dashed line. The results clearly show that the proposed scheme obtains a very high correlation for near or far data for most of the considered range, i.e., between 40 and 360 cm. Near correlation is close to one until 80 cm and quickly drops to zero by 120 cm; conversely, far correlation is initially very low and then increases to more than 0.9 between 120 and 340 cm. Far correlation decreases with distance after 300 cm due to the noise in the captured image, eventually resulting in unrecoverable errors. Nevertheless, the results clearly demonstrate the separation obtained in practice: the strong correlation with near data at 100 cm is followed by a strong correlation with far data just 40 cm farther away.

Figure 6b shows the barcode decoding success as a function of the distance for different media. The ranges with non-zero decoding success are grouped together for near and far data, and the average value is shown under the corresponding bars; the range between them with no successful scans (i.e., neither near nor far) is also explicitly indicated. The results clearly show how the separation of distance-dependent barcodes is consistent across the different media, with only a small gap between successful near and far scans. The reliability is also very similar, with average values always above an 89% success rate at both near and far distance. Clearly, the specific ranges vary based on the size of the barcode: the smaller one can be successfully employed until at least 90 cm, while the larger barcode reaches up to 3 m. This shows how distance-dependent barcodes can be dimensioned to suit different needs in diverse use cases. Note that the separation for the smaller barcodes roughly corresponds to that of the larger ones upon rescaling them by a factor of 2.5, namely, the same value as the ratio between the size of the two considered barcodes.

Figure 6c shows the decoding success as a function of the distance for different phones for a 16.5 cm barcode shown on an IPS display.



(a) (b) (c)

Figure 6: (a) Separation between near and far decoding in terms of byte correlation with source data as a function of distance. Reliability in terms of successfully decoded barcodes as a function of distance by target zone (i.e., near and far) for (b) different media and (c) different phones.

The figure shows that the range obtained[4] with an iPhone 7 is lower than that of the OnePlus 6. This is expected, as the phone has a 12 MP camera. Nevertheless, the reliability is generally comparable. This shows the impact of the optics and of the image sensor in addition to the sheer number of megapixels. In contrast, the Nokia 8 phone (with a 13 MP camera) has a slightly longer range, but a lower average success rate at farther distances, together with a higher standard deviation than in the rest of the results. This happens as the reliability drops to 40% at 220 cm; in contrast, the success in the range between 120 and 240 cm is always above 80%. Most importantly, separation occurs with all phones: successful scans consist of either near or far data for any considered distance.

Comparison with state of the art. The second set of experiments aims at comparing the effectiveness of the proposed distance-dependent barcode with the state of the art, namely, the QR code-based and FOCUS-based schemes (described in Section 5.1). Figure 7 shows the reliability of the different solutions obtained by scanning the respective barcodes at different distances.

Figure 7a shows the percentage of successfully decoded near and far frames for distance-dependent barcodes. The results demonstrate how the proposed scheme is effective: near data are always successfully decoded in a short range (40-80 cm), while far data has consistently high success rate at larger distances – even 100% between 140 and 320 cm. Similar to Figure 6a, the plot shows a clear separation between successful near and far scans. After 320 cm, reliability decreases due to noise and the resulting decoding errors cannot be recovered by forward error correction. Nevertheless, the distance-dependent barcodes obtain a maximum scanning range of at least 3 meters, which is practical for most applications (as those presented in Section 3).

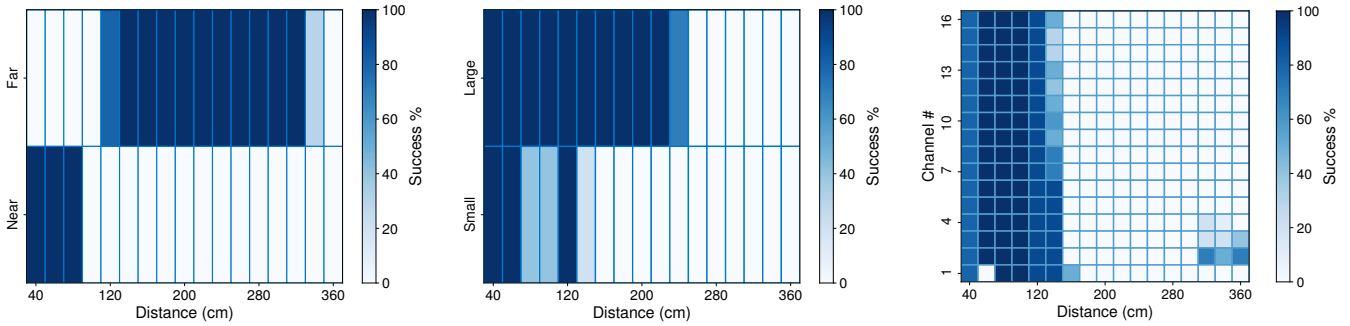
Figure 7b shows the decoding success for the QR code-based scheme. The figure shows the obtained results for both the small and large QR codes, wherein colors correspond to the success rates reported on the right side of the plot. The small QR code can be successfully scanned only in the range between 40 to 140 cm, although with highly varying reliability (i.e., between 20% and 100%) which depends on the lighting conditions in the experiments. The large QR code can be reliably decoded (i.e., with no less than 70% success) until 240 cm with a very consistent success rate. The maximum achievable range of 240 cm is lower than the one obtained by the distance-dependent barcodes. This can be explained by the intrinsic nature of QR codes, which are primarily meant to be scanned from close proximity. In particular, the corner locators and the timing patterns are less and less robust to noise when the block size becomes smaller.

Figure 7c shows the decoding success for the FOCUS-based scheme. The figure reports the obtained results for the individual channels in the FOCUS barcode on the vertical axis, where channels with smaller numbers correspond to lower frequencies. The results show how channels in lower frequencies obtain some higher success over longer distances. However, the longer range obtained for lower frequencies does not correspond to consistently high decoding success. For instance, data in channel 2 has a success rate over 80% only until 140 cm. Moreover, the distance of successful scans does not significantly increase as the channel number (i.e., the frequency) decreases. This not only limits the maximum scanning range, but also reduces separation between near and far data. This result is due to the peculiar code construction in FOCUS: channels therein are not spatially separated, but rather partitioned into distinct yet contiguous frequency ranges. Consequently, reliability is largely affected by decoding errors close to the frequency boundaries of adjacent channels. Moreover, FOCUS encodes data into a grayscale image, which makes it more sensitive to noise (including the moiré effect), especially in the higher frequencies.

Summary and discussion. The experimental results demonstrate that the proposed distance-dependent barcode design is effective: it achieves a clear separation between near and far data with high reliability. The consistency and robustness of the proposed approach is also demonstrated by the experiments carried out by using different display types, barcode sizes, and smartphones. The results clearly show how the encoding is not affected by the actual panel type.

Distance-dependence could also be achieved by using two QR codes with different sizes at the cost of reducing the usable area of the considered medium. Note that the simultaneous presence of two

[4]For simplicity, the images captured by the iPhone are given as input to the Android implementation of the scanner.



(a) (b) (c)

Figure 7: Reliability as percentage of successfully decoded frames broken down by type (i.e., near and far) for (a) distance-dependent, (b) QR code-based and (c) FOCUS-based barcodes.

barcodes entails serious practical issues. For instance, widely-used QR code scanners cannot reliably (if at all) detect the presence of multiple barcodes; asking the user to manually focus the image onto either the near or far barcode is clearly not practical. Finally, FOCUS-based barcodes do not allow to obtain both high separation and high reliability, despite the availability of multiple frequency channels. Clearly, FOCUS was not designed for the use case considered here, as opposed to streaming data over screen-camera links. Nevertheless, the proposed scheme based on color blending is simpler yet effective.

6 USER STUDY

In addition to the performance evaluation above, a user study was conducted to evaluate how the proposed scheme is able to support a representative mobile application from the user's perspective.

6.1 Setup and methodology

The user study was conducted for the use case of casual interactions with public displays[5] (presented in Section 3.1) to ensure that participants had to explicitly scan a distance-dependent barcode. For this purpose, a web-based presentation system was realized, similar to the one in [19]. The distance-dependent barcode is implemented in JavaScript and embeds a URL with a token that is different for near and far viewers. Once scanned, such a URL determines the role of the user: the first near viewer to scan an available display becomes the presenter; all far viewers become audience. The realized system follows the client-server architecture and employs the RevealJS presentation framework for rich HTML-based presentations.

Participants were selected to include a pool of potential users with different backgrounds. In total, 30 participants (21 male and 9 female) were involved in the user study, including people working in the authors' institution as well as visitors; their age ranged between 25 and 54 years. The participants volunteered under informed consent and received a movie ticket as an incentive for participation. The study was designed as a controlled experiment; the participants were briefly introduced to the use case and the intended task beforehand. The experiment was conducted in a meeting room with both natural and artificial light. The experimental setup included a 55" LG Ultra HD TV with a 4K IPS display connected to a personal computer (as in Figure 2a) and a OnePlus 6 smartphone with the distance-dependent barcode scanner preinstalled. Participants were instructed to scan the barcode shown on the display with the smartphone at different positions – clearly marked in the room as *near* and *far* – and perform different actions based on the obtained result. During the experiment, participants were encouraged to express their thoughts and impressions when using the application according to the think-aloud protocol; their observations were recorded for further analysis. After the experiment was over, the participants were asked to complete an online questionnaire. The questionnaire covered aspects related to the application in the user study as well as previous experience with related technologies (e.g., QR codes). To evaluate user satisfaction, a 5-point Likert scale was used, wherein 1 corresponds to "Strongly disagree" and 5 to "Strongly agree".

6.2 Findings

The following summarizes the results of the user study, categorized according to their most important dimensions.

Usability. Most of the participants found the application *easy to use*, *pleasant to use*, and *efficient*: the median rating was 4 for all these aspects[6] in the questionnaire as shown in Figure 8. This was probably because all the participants had previous experience with barcodes, specifically, in scanning them with mobile devices. The barcode was correctly decoded at the first try for both distances in most cases. Three participants failed to successfully scan the barcode from the far distance and attempted to zoom the camera preview to better fit the barcode, which was not supported by the application. The subsequent scan was indeed successful in all three cases.

Understanding. When asked about the use of the barcode, most of the participants understood the purpose of scanning the barcode. One user mentioned *"I liked that it didn't require any training i.e.*

[5] An important aspect of the study is to relate user experience to the familiar process of scanning QR codes, while the specifics of the application itself are not central to the evaluation.

[6] The same wording as the italicized text was used in the questionnaire.

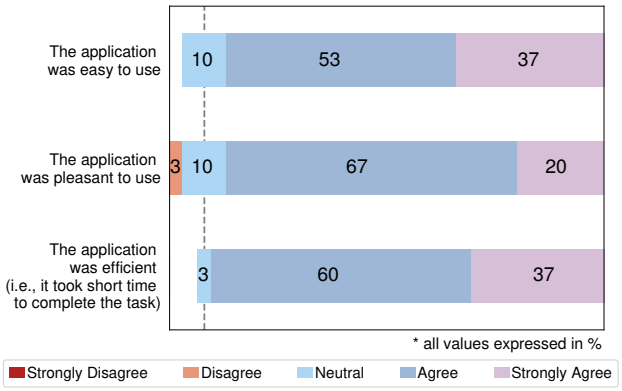


Figure 8: Questions in the user study and Likert-scale responses (N=30).

even without knowing how it's supposed to work, it would be used just as a regular barcode [...]"; another said *"I didn't have to train myself. It was very intuitive"*. However, a few users felt that the barcode did not look as one to be scanned. One user mentioned *"The code didn't look to me as a code so I wouldn't know that it was supposed to be scanned if seen somewhere else"*, another one *"If it's not written to scan the barcode, it will be difficult to understand if it's a barcode or not"*. Moreover, the participants did not realize that the barcode provided different data depending on the distance. One user commented *"Near and far are not obvious without markers or more information"*. Another participant said *"without instructions I wouldn't know it has different results depending on the distance"*. These considerations highlight two important aspects. First, using distance-dependent barcodes is *transparent* to the users; the familiar process of mobile barcode scanning applies here as well, and the context-dependent result is not even noticeable. Second, users need explicit *cues* on how distance-dependent barcodes apply to a specific use case. They include a clear explanation on the outcome of scanning such barcodes and how to distinguish between near and far locations. This could be accomplished by adding visual hints to the physical environment, for instance, separation lines for near-far zones or other types of markers – as indeed done in the user study, similar to [5, 32].

Distance-dependence. Overall, the participants were very satisfied with the distance-dependence provided by the barcodes. One participant mentioned *"The idea was totally new to me. Never thought that it's possible to interpret a barcode in two different ways depending on the distance"*. Another participant reported *"The near and far way of encoding information seems pretty useful and seems to be a smart generalization of barcodes"*. Furthermore, the users found the proposed barcode particularly beneficial for the considered use case. One user mentioned *"It's flexible and the same barcode can be used both by the speaker and the audience. Speaker can control switching of the slides, which is shown on the audience's phones in real time. People in the audience have contents of the slides on their phones for later use"*; another one *"I think, it would be the easiest way to pass control of the slides from one person to other. The best part is everyone can control from their own mobile device"*.

7 CONCLUSION

This article introduced the novel concept of *distance-dependent barcodes*, which provide users with different data depending on their distance (i.e., near or far). The key method to achieve this is the design of a distance-dependent encoding based on color blending that works for diverse media, including paper and different types of displays. The proposed solution was implemented for Android-based devices and was evaluated in a variety of conditions. The obtained results demonstrate that distance-dependent barcodes are robust and provide a clear separation between near and far scanners. This makes the proposed solution especially practical for context-aware mobile applications; among them, casual interactions with public displays and augmented reality for retail were considered as representative use cases. Distance-dependent barcodes were also shown to be user-friendly and effective through a user study.

On the one hand, reliability is primarily affected by image compression, especially at the boundaries of the near and far ranges. This issue is difficult to overcome, also because uncompressed images are large and unsupported by most off-the-shelf smartphones. A preliminary investigation on using additional colors has shown a limited improvement in capacity, as it becomes significantly more challenging to reliably decode the data due to increased errors. On the other hand, separation largely depends on the barcode size and the device-specific capture resolution. This work has considered only a binary near / far separation through modules with simple macro blocks and patterns. However, a higher number of sub-blocks and more complex shapes in the patterns could also be employed, for instance, to support multiple distance levels or to improve performance. These aspects could be addressed in a future study.

Finally, it would be interesting to investigate machine-learning methods that train the decoder based on extensive experimental data to further improve the decoding success. Analytical modeling of color blending in realistic scenarios would also be a promising research direction.

ACKNOWLEDGMENTS

This work was partially supported by the Academy of Finland under grants number 299222 and 332307. The authors would like to thank: Jaakko Lehtinen, Antti Oulasvirta, and Tuomas Aura for their feedback; the CSC – IT Center for Science for provisioning the computational resources used in the study.

REFERENCES

[1] Satoshi Abe, Takefumi Hiraki, Shogo Fukushima, and Takeshi Naemura. 2018. Screen-Camera Communication via Matrix Barcode Utilizing Imperceptible Color Vibration. In *The 31st Annual ACM Symposium on User Interface Software and Technology Adjunct Proceedings* (Berlin, Germany) *(UIST '18 Adjunct)*. Association for Computing Machinery, New York, NY, USA, 166–168. https://doi.org/10.1145/3266037.3271638

[2] Wilhelm Burger and Mark J Burge. 2016. *Digital image processing: an algorithmic introduction using Java*. Springer.

[3] Dimitris Chatzopoulos, Carlos Bermejo, Zhanpeng Huang, and Pan Hui. 2017. Mobile Augmented Reality Survey: From Where We Are to Where We Go. *IEEE Access* 5 (2017), 6917–6950. https://doi.org/10.1109/ACCESS.2017.2698164

[4] Brian X. Chen. 2019. The Reason Your Photos Are About to Get a Lot Better. https://www.nytimes.com/2019/10/15/technology/personaltech/google-pixel-photography.html.

[5] Jakub Dostal, Uta Hinrichs, Per Ola Kristensson, and Aaron Quigley. 2014. SpiderEyes: Designing Attention- and Proximity-aware Collaborative Interfaces for Wall-sized Displays. In *Proceedings of the 19th International Conference on Intelligent User Interfaces (IUI '14)*. 143–152. https://doi.org/10.1145/2557500.

2557541

[6] Gonzalo J. Garateguy, Gonzalo R. Arce, Daniel L. Lau, and Ofelia P. Villarreal. 2014. QR Images: Optimized Image Embedding in QR Codes. *IEEE Transactions on Image Processing* 23, 7 (July 2014), 2842–2853. https://doi.org/10.1109/TIP.2014.2321501

[7] Tian Hao, Ruogu Zhou, and Guoliang Xing. 2012. COBRA: color barcode streaming for smartphone systems. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*. ACM, 85–98.

[8] Frederik Hermans, Liam McNamara, Gábor Sörös, Christian Rohner, Thiemo Voigt, and Edith Ngai. 2016. FOCUS: Robust Visual Codes for Everyone. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services* (Singapore, Singapore) *(MobiSys '16)*. 319–332. https://doi.org/10.1145/2906388.2906399

[9] Steve Hranilovic and Frank R. Kschischang. 2006. A pixelated MIMO wireless optical communication system. *IEEE Journal of Selected Topics in Quantum Electronics* 12, 4 (July 2006), 859–874. https://doi.org/10.1109/JSTQE.2006.876601

[10] Wenjun Hu, Jingshu Mao, Zihui Huang, Yiqing Xue, Junfeng She, Kaigui Bian, and Guobin Shen. 2014. Strata: Layered Coding for Scalable Visual Communication. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking* (Maui, Hawaii, USA) *(MobiCom '14)*. 79–90. https://doi.org/10.1145/2639108.2639132

[11] Wenjian Huang and Wai Ho Mow. 2013. PiCode: 2D barcode with embedded picture and ViCode: 3D barcode with embedded video. In *Proceedings of the 19th annual international conference on Mobile computing & networking*. ACM, 139–142.

[12] Hiroko Kato and Keng T. Tan. 2007. Pervasive 2D Barcodes for Camera Phone Applications. *IEEE Pervasive Computing* 6, 4 (Oct 2007), 76–85. https://doi.org/10.1109/MPRV.2007.80

[13] Hiroko Kato, Keng T Tan, and Douglas Chai. 2010. *Barcodes for mobile devices*. Cambridge University Press.

[14] Ali Khalajmehrabadi, Nikolaos Gatsis, and David Akopian. 2017. Modern WLAN Fingerprinting Indoor Positioning Methods and Deployment Challenges. *IEEE Communications Surveys Tutorials* 19, 3 (thirdquarter 2017), 1974–2002. https://doi.org/10.1109/COMST.2017.2671454

[15] Kai Kuikkaniemi, Vilma Lehtinen, Matti Nelimarkka, Max Vilkki, Jouni Ojala, and Giulio Jacucci. 2014. Designing for Presenters at Public Walk-up-and-use Displays. In *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction (TEI '14)*. 225–232. https://doi.org/10.1145/2540930.2540949

[16] Jim F. Kurose and Keith W. Ross. 2017. *Computer Networking: A Top-Down Approach* (7th ed.). Pearson.

[17] Tobias Langlotz and Oliver Bimber. 2007. Unsynchronized 4D barcodes. In *International Symposium on Visual Computing*. Springer, 363–374.

[18] Akira Matsumoto, Satoshi Abe, Takefumi Hiraki, Shogo Fukushima, and Takeshi Naemura. 2019. Imperceptible AR Markers for Near-Screen Mobile Interaction. *IEEE Access* 7 (2019), 79927–79933. https://doi.org/10.1109/ACCESS.2019.2921580

[19] Elena Oat, Mario Di Francesco, and Tuomas Aura. 2014. MoCHA: Augmenting Pervasive Displays through Mobile Devices and Web-based Technologies. In *The 1st IEEE Workshop on Developing Applications for Pervasive Display Networks (PD-Apps '14)*. 506–511. http://uc.inf.usi.ch/sites/all/files/pd-apps2014/papers/1569868391.pdf

[20] Aude Oliva, Antonio Torralba, and Philippe G. Schyns. 2006. Hybrid Images. *ACM Trans. Graph.* 25, 3 (July 2006), 527–532. https://doi.org/10.1145/1141911.1141919

[21] Samuel David Perli, Nabeel Ahmed, and Dina Katabi. 2010. PixNet: interference-free wireless links using LCD-camera pairs. In *Proc. of MobiCom '10*. 137–148.

[22] Nathan Piasco, Désiré Sidibé, Cédric Demonceaux, and Valérie Gouet-Brunet. 2018. A survey on Visual-Based Localization: On the benefit of heterogeneous data. *Pattern Recognition* 74 (2018), 90–109. https://doi.org/10.1016/j.patcog.2017.09.013

[23] Marco Querini, Antonio Grillo, Alessandro Lentini, and Giuseppe F. Italiano. 2011. 2D Color Barcodes for Mobile Phones. *IJCSA* 8, 1 (2011), 136–155.

[24] Irving S. Reed and Gustave Solomon. 1960. Polynomial Codes Over Certain Finite Fields. *Journ. Soc. for Ind. and Applied Mathematics* 8, 2 (1960), 300–304. https://doi.org/10.1137/0108018

[25] Microsoft Research. 2009. High Capacity Color Barcode Technology.

[26] Theresa-Marie Rhyne. 2016. *Applying color theory to digital media and visualization* (first ed.). CRC Press.

[27] Russ Cox. Retrieved November 27, 2018. QArt Codes. https://research.swtch.com/qart.

[28] Mohit Sethi, Elena Oat, Mario Di Francesco, and Tuomas Aura. 2014. Secure Bootstrapping of Cloud-Managed Ubiquitous Displays. In *The 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp 2014)*. 739–750. https://doi.org/10.1145/2632048.2632049

[29] Mona Singh and Munindar P. Singh. 2013. Augmented Reality Interfaces. *IEEE Internet Computing* 17, 6 (Nov 2013), 66–70. https://doi.org/10.1109/MIC.2013.107

[30] Jesús Téllez and Sherali Zeadally. 2017. *Mobile Payment Systems*. Springer.

[31] Robert Ulichney. 1987. *Digital halftoning*. MIT press.

[32] Mélodie Vidal, Andreas Bulling, and Hans Gellersen. 2013. Pursuits: Spontaneous Interaction with Displays Based on Smooth Pursuit Eye Movement and Moving Targets. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '13)*. 439–448. https://doi.org/10.1145/2493432.2493477

[33] Anran Wang, Shuai Ma, Chunming Hu, Jinpeng Huai, Chunyi Peng, and Guobin Shen. 2014. Enhancing Reliability to Boost the Throughput over Screen-camera Links. In *The 20th Annual International Conference on Mobile Computing and Networking* (Maui, Hawaii, USA) *(MobiCom '14)*. 41–52. https://doi.org/10.1145/2639108.2639135

[34] Qian Wang, Man Zhou, Kui Ren, Tao Lei, Jikun Li, and Zhibo Wang. 2015. Rain Bar: Robust Application-Driven Visual Communication Using Color Barcodes. In *2015 IEEE 35th International Conference on Distributed Computing Systems*. 537–546. https://doi.org/10.1109/ICDCS.2015.61

[35] Roy Want, Bill N. Schilit, and Scott Jenson. 2015. Enabling the Internet of Things. *Computer* 48, 1 (January 2015), 28–35. https://doi.org/10.1109/MC.2015.12

[36] Zhe Yang, Yuting Bao, Chuhao Luo, Xingya Zhao, Siyu Zhu, Chunyi Peng, Yunxin Liu, and Xinbing Wang. 2016. ARTcode: Preserve Art and Code in Any Image. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing* (Heidelberg, Germany) *(UbiComp '16)*. 904–915. https://doi.org/10.1145/2971648.2971733

[37] Takuro Yonezawa, Masaki Ogawa, Yutaro Kyono, Hiroki Nozaki, Jin Nakazawa, Osamu Nakamura, and Hideyuki Tokuda. 2014. SENSeTREAM: Enhancing Online Live Experience with Sensor-federated Video Stream Using Animated Two-dimensional Code. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '14)*. 301–305. https://doi.org/10.1145/2632048.2632104

[38] Chungkuk Yoo, Inseok Hwang, Seungwoo Kang, Myung-Chul Kim, Seonghoon Kim, Daeyoung Won, Yu Gu, and Junehwa Song. 2017. Card-stunt As a Service: Empowering a Massively Packed Crowd for Instant Collective Expressiveness. In *Proceedings of the 15th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '17)*. 121–135. https://doi.org/10.1145/3081333.3081357

[39] Tong Zhan, Wenzhong Li, Xu Chen, and Sanglu Lu. 2018. Capturing the Shifting Shapes: Enabling Efficient Screen-Camera Communication with a Pattern-based Dynamic Barcode. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 52 (March 2018), 25 pages. https://doi.org/10.1145/3191784

[40] Tong Zhan, Wenzhong Li, Xu Chen, and Sanglu Lu. 2019. MegaLight: Learning-based Color Adaptation for Barcode Stream Recognition over Screen-Camera Links. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 2, Article 66 (June 2019), 23 pages. https://doi.org/10.1145/3328937