JARS Journal of Applied Remote Sensing

RESEARCH PAPER

# Feasibility of remote estimation of optical turbulence via quick response code imaging

Burton Neuner III, Skylar D. Lilledahl, and Kyle R. Drexler*
Naval Information Warfare Center Pacific, San Diego, California, United States

**ABSTRACT.** Turbulence estimation theory is presented and demonstrated by imaging a series of spatially encoded quick response (QR) codes in ambient radiation through atmospheric scintillation. This remote sensing concept was verified though preliminary feasibility experiments and detailed MATLAB simulations using QR codes displayed on a low-power digital e-ink screen. Of note, knowledge of propagation range and QR code dimensions are not required ahead of time, as each code contains information detailing its block size and overall physical size, enabling automated calculations of spatial resolution and target range. Estimation algorithms leverage the extracted resolution and range information to determine path-integrated optical turbulence, as quantified by the Fried parameter, $r_0$. The estimation criterion is obtained by cycling a series of QR code sizes on an e-ink screen and determining the transition point at which the QR code can no longer be read, resulting in a system capable of automatically estimating path-integrated optical turbulence.

**Keywords:** turbulence; scintillation; atmospheric propagation; imaging; remote sensing; QR code

Paper 230443G received Aug. 18, 2023; revised Dec. 1, 2023; accepted Dec. 27, 2023; published Jan. 18, 2024.

## 1 Introduction

Atmospheric turbulence degrades long-range imaging and blurs important target features.[1–6] Real-time knowledge of turbulence and target distance are valuable to many industries and research organizations, especially during test and evaluation events. Turbulence measurements are inherently difficult, and ranging measurements typically require radio transceivers, multiple cameras in stereo imaging configurations, or light detection and ranging (LIDAR) devices. Systems to measure and estimate turbulence include meteorological nodal measurement techniques, path-integrated techniques, and numerical weather modeling. Nodal turbulence measurement systems, such as high-speed temperature and sonic anemometer sensors, are only accurate at a given geographical point.

Complex numerical weather models are able to estimate environmental properties, and they infer optical property estimates, including optical turbulence. These models are heavily reliant on the planetary boundary layer (PBL), which can be measured using LIDAR. Knowledge of this turbulent layer height improves weather forecast models.[7] Another study on PBL models using turbulent layer mixing and weather forecasting leveraged parallel computing.[8] However, such models often require significant computational power that make them prohibitively expensive, both in time and hardware. Furthermore, error estimates are difficult to determine.

Path-integrated optical sensors can directly measure turbulence, but typical systems require an active source (e.g., laser) at one end and powered electronics at both ends of the propagation

*Address all correspondence to Kyle R. Drexler, kyle.r.drexler.civ@us.navy.mil range. Active sensors also have the additional complication of path weighting, where a specific portion of the path will have an increased effect on the measurement. This may or may not be advantageous to a specific test and evaluation event. Path-integrated optical sensing also finds applications in underwater imaging through turbulence. In one technique, machine learning algorithms and multi-frame image fusion predict and correct for underwater image degradation.[9] In another technique, Gaussian beams carrying orbital angular momentum and unique modulation schemes can be leveraged to characterize the strength and length scales of underwater optical phase distortions.[10] Real-time independent estimates of optical turbulence could further enhance these innovative techniques.

Passive systems leverage ambient radiation for optical analysis. Distance can be estimated by imaging passive 2D pseudo random test targets and quick response (QR) codes through a blurring phase mask.[11] While this technique can restore degraded image quality and estimate distance, its use is limited to indoor distances (currently, on the order of meters), and the blur metric must be known ahead of time. Estimating range from QR codes has been demonstrated using Fresnel zone aperture lensless imaging and image sharpness evaluation.[12] QR codes displayed on LED screens enable precise 2D indoor positioning.[13] High-precision mobile robot positioning and docking are enabled by ground-based passive QR code arrays,[14] and 6 degree of freedom ground truthing is possible using AprilTag codes.[15] These techniques are accurate, but they are also limited to modest distances, and they do not enable turbulence estimation.

This work presents a new method to estimate path-integrated optical turbulence and target range by imaging spatially encoded QR codes. The technique leverages standard barcode-reading software, requires minimal computational power, and eliminates the need for bespoke machine learning algorithms. Fundamentally, a QR code is a binary black and white image that represents information within a square pattern composed of black and white square blocks. Prior work demonstrated that QR codes are useful targets when quantifying imaging performance by leveraging pass/fail metrics from a standard barcode reader.[16–20] This work extends the use of QR codes to turbulence quantification for more generalized applications using a standard imaging camera.

Embedding the physical properties of the QR code into the encoded information provides the optical imaging system inherent knowledge of both propagation range and spatial resolution to enable turbulence estimation. The technique is realized by imaging and extracting information from a series of QR codes of increasing or decreasing sizes to detect the smallest resolvable code. Once the transition point is identified, the minimum estimated spatial resolution is known. The series of QR codes can be static (e.g., all codes printed on a target board) or dynamic (e.g., cycling in time on a low-power digital screen).

The remainder of this paper presents the mathematical details along with experiments and simulations to support the concept of remotely estimating path-integrated optical turbulence using spatially encoded QR codes.

## 2 Theoretical Approach

Atmospheric turbulence arises from temperature and pressure changes that induce fluctuations of the refractive index. This natural phenomenon is detailed in Refs. 1–6. and is summarized below. The atmospheric index of refraction, $n$, is defined in Eq. (1), where $n_0$ is the average index of refraction, $n_1$ is the randomly fluctuating term, $r$ is the 3D position vector, and $t$ is time

$$n(r,t) = n_0 + n_1(r,t). \tag{1}$$

To define the statistical nature of atmospheric turbulence, Kolmogorov theory is employed,[2] defining the spatial power spectral density of the fluctuating term $n_1$ in Eq. (1), where $C_n^2(h)$ is the refractive-index structure parameter at altitude $h$; $\lambda$ is the optical wavelength; and $k$ is the optical wavenumber (with $k = 2\pi/\lambda$)

$$\Phi_n^k(k,h) = 0.033C_n^2(h)k^{-\frac{11}{3}}. \tag{2}$$

In simulation studies, the refractive-index structure parameter is a driving force for modeling atmospheric turbulence, with common values ranging from $10^{-18}$ m$^{-2/3}$ (for minimal turbulence effects) to $10^{-12}$ m$^{-2/3}$ (for strong turbulence effects). Atmospheric turbulence is inherently a zero-mean process. From an imaging perspective, this means the centroid of a light ray will average out and higher frequency information will be lost. Fried[1] first quantified atmospheric seeing by introducing the Fried atmospheric coherence length parameter, $r_0$, where

$$r_0 = 0.185\left[\frac{4\pi^2}{k^2 C_n^2 z}\right]^{\frac{3}{5}}, \tag{3}$$

and $z$ is the propagation distance in meters.

In a vacuum, the far-field diffraction pattern's angular extent (or seeing angle) is $\lambda/D$, where $D$ is the optical entrance diameter. In the atmosphere, the turbulence seeing angle is limited by Fried's parameter as $\lambda/r_0$.[21] Larger entrance diameters indicate better diffraction-limited resolution, and larger values of $r_0$ indicate better atmospheric conditions. Spatial resolution of the target, $\Delta x$, is a function of the seeing angle (the larger of $\lambda/D$ and $\lambda/r_0$) and the propagation distance, $z$. As a result, the overall spatial resolution will be constrained by the diffraction-limited optical system ($D$) or by the extent of turbulence ($r_0$): the smaller of these two parameters will limit the spatial resolution of the imaging system as

$$\Delta x = \frac{\lambda z}{\min([D, r_0])}. \tag{4}$$

From a resolution perspective, when collecting image data, the $r_0$ parameter effectively reduces the input aperture of the imaging system. Conceptually, and from a simplified modeling and simulation perspective, it can be thought of as applying a low-pass filter. Typical values for Fried's parameter range from about 2 to 15 cm (0.8 inches to 6.0 inches) at sea level for the visible band. To enable an estimation of $r_0$ in a passive imaging experiment, the entrance aperture must be larger than the maximum expected $r_0$ value. For characterization purposes, a maximum aperture diameter of 15 cm (∼6 inches) is sufficient for most sea-level conditions.

Fried's parameter can be estimated from the spatial resolution of Eq. (4), which assumes optical diameters sufficient to study mild to severe turbulence, such that $\min([D, r_0]) = r_0$. Thus, Fried's parameter is estimated as

$$r_{0,\mathrm{est}} = \frac{\lambda z}{\Delta x}. \tag{5}$$

In this work, the median wavelength within the visible band is set to $\lambda = 550$ nm. Spatial resolution $\Delta x$ is taken as the smallest possible QR code block size that is first detected by the imaging system and scanning algorithm. Propagation range $z$ is estimated by assuming an optical proportionality where the ratio of the QR code object size ($\mathrm{QR}_o$) to propagation range ($z$) is equal to the ratio of the QR code image size ($\mathrm{QR}_i$) to telescope focal length ($f_l$). As such, range is estimated as

$$z = f_l \frac{\mathrm{QR}_o}{\mathrm{QR}_i}. \tag{6}$$

QR code object size, $\mathrm{QR}_o$, is the physical height of the code displayed on the e-ink screen. QR code image size, $\mathrm{QR}_i$, is the physical height of the code projected onto the camera sensor. After the camera sensor receives the QR code image and the system records the file, the code size ($\mathrm{QR}_i$) is inherently measured by the scanning algorithm that looks for key features embedded within the code. With estimated values for spatial resolution and propagation range, Fried's parameter is then automatically calculated for each successfully read QR code using Eq. (5), which is restated as

$$r_{0,\mathrm{est}} = \lambda f_l \frac{\mathrm{QR}_o}{\mathrm{QR}_i}\frac{1}{\Delta x}. \tag{7}$$

For this feasibility study, impacts to the turbulence estimate and quantified uncertainty were not explored for the numerous possible QR code encoding schemes, which include version number and error correction (EC) level. This study is limited to a single, practical QR code version to simplify the analysis by maintaining the same block density for all images. In this way, the only significant parameter that changes between images is the physical size of the block, which equates to imaging resolution, $\Delta x$.

Additionally, QR codes are created to maximize a uniform distribution of black and white module blocks, and this is done within the code generator by applying one of the eight available masks that optimize uniformity. This inherent block uniformity, when combined with the inherent error correction within the QR encoding mechanism, minimizes turbulence estimation anomalies between different codes. With the theoretical approach for estimating $r_{0,\text{est}}$ complete, the technical approach for implementing this method follows.

## 3 Technical Approach

### 3.1 Hardware

To demonstrate the concept of remote turbulence sensing, spatially encoded QR codes were created for use with a digital e-ink screen. As seen in Fig. 1, this study employed a 6-inch (152.4-mm-diagonal) ePaper screen (model ED060KC1) manufactured by E Ink; it is referred to as the e-ink screen in this paper. This low-cost device is small, lightweight, and consumes minimal power. The screen is 1448 pixels wide by 1072 pixels tall. With a square pixel pitch of 0.0845 mm, the resulting screen size is 122.36 mm wide by 90.58 mm tall.

For realistic simulations of atmospheric propagation, a commercial telescope provided relevant optical imaging specifications. The Orion CC6 Classical Cassegrain 6-inch telescope (model 52138) has an optical diameter of $D = 152.4$ mm, a focal length of $f_l = 1836$ mm, and a focal ratio of $f/12.0$. As described in prior publications,[19,20] the authors use a commercial camera for laboratory experiments; the specifications for this camera provide relevant values for the MATLAB simulations. The FLIR Blackfly S USB3 camera (model number BFS-U3-32S4M-C) features the Sony IMX252 monochrome sensor. This sensor is 2048 pixels wide by 1536 pixels tall. With a square pixel pitch of 3.45 $\mu$m, the resulting sensor size is 7.07 mm wide by 5.30 mm tall. With hardware defined for the initial feasibility experiments and principal quantitative simulations, the task of generating spatially encoded QR codes followed.

### 3.2 QR Code Generation

The initial proof of concept used five unique codes[22] to span the e-ink screen's height scale. The smallest used 9.3% of the screen height while the largest used 92.4% of the screen height. The following properties were considered to optimize turbulence estimation performance: QR code version, EC level, encoding alphabet, data payload size (i.e., string length), and data parsing (i.e., delimiting characters and fixed widths).

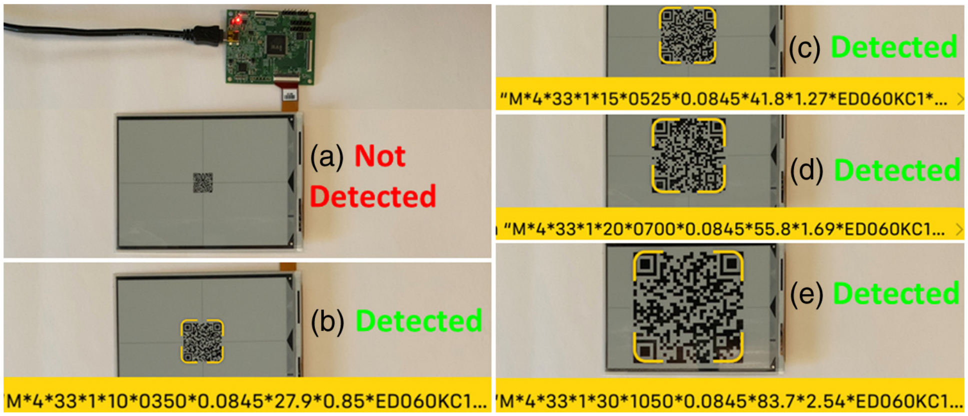

(a) Not Detected
(b) Detected
'M*4*33*1*10*0350*0.0845*27.9*0.85*ED060KC1...
(c) Detected
"M*4*33*1*15*0525*0.0845*41.8*1.27*ED060KC1*...
(d) Detected
"M*4*33*1*20*0700*0.0845*55.8*1.69*ED060KC1...
(e) Detected
'M*4*33*1*30*1050*0.0845*83.7*2.54*ED060KC1...

Fig. 1 Smartphone images show results from feasibility experiments. The e-ink screen cycles from smaller to larger images while the smartphone scanner seeks QR codes. The scanner fails to detect the M05 code in panel (a), as code dimensions are below the optical resolution. However, the scanner detects the M10 code in panel (b) and immediately displays the character string. This "detected" metric indicates resolvable codes, which contain QR code height and block size within the strings. Subsequent detections of larger codes (c) M15, (d) M20, and (e) M30 are then ignored.

The QR code version defines the number of blocks that comprise the code. As the version number increases, data density and block density also increase. As a result, data storage capacity and block size are inversely proportional for a fixed QR code size. In this work, version 4 (V4) codes provide the ideal balance between maximal string length and maximal block size: these QR codes are 33 blocks wide by 33 blocks tall. The QR code generator applied a border margin of one white block around each code; while this white block margin is not observed visually, it is accounted for when sizing images for display on the e-ink screen.

There are four levels of EC available when generating QR codes: low (L), medium (M), quartile (Q), and high (H).[23] These levels correspond to data redundancy of 7%, 15%, 25%, and 30%, respectively. For a given QR code version, lower EC levels permit more encoded characters, but such codes are less robust against pattern degradation. Conversely, higher EC levels permit fewer characters, but such codes are more robust against degradation. In this work, level-M EC was used to balance these features.

When encoding data within a QR code, an alphabet must be selected. Three examples explored include the numeric, alphanumeric, and ISO-8859-1 alphabets. Numeric encoding is limited to the 10 digits (0-9), but it allows for the largest data payloads. Alphanumeric encoding supports 45 characters (which includes capitalized letters, numbers, and limited special characters), and it allows for moderate data payloads. ISO-8859-1 encoding supports 191 characters, but it offers the smallest data payloads. Alphanumeric encoding was selected to optimize both readability and data payload size. With QR Code V4 and level-M error correction, the alphanumeric alphabet allows for a maximum of 90 characters. This work features codes with 43 data characters and 30 appended padding characters for a total of 73 characters. The padding characters allow for future data payload expansion without changes to code version or EC level.

Character strings are parsed by machine code using delimiter symbols and/or fixed widths. For future flexibility, the strings presented in this work leverage both: the asterisk symbol (*) is the delimiter between data fields, and fixed widths maintain consistent data fields. To ensure widths remained constant, leading zeros padded numerical values where necessary. Each of the five QR codes contain the following eleven data fields.

1. EC level, which is fixed at $M$.
2. QR code version, which is fixed at 4.
3. QR code height in units of blocks, which is fixed at 33.
4. QR code white margin in units of blocks, which is fixed at 1.
5. QR code block height in units of pixels, with the following options: 05, 10, 15, 20, and 30. These options correspond to the five codes, from smallest to largest. All of the following five-field options below match this order.
6. QR code image file height (which includes the margin) in units of pixels, with the following options: 0175, 0350, 0525, 0700, and 1050.
7. e-ink screen pixel size in units of mm, which is 0.0845.
8. QR code height visible on the e-ink screen (which excludes the margin) in units of mm, with the following options: 13.9, 27.9, 41.8, 55.8, and 83.7.
9. QR code block height displayed on the e-ink screen in units of mm, with the following options: 0.42, 0.85, 1.27, 1.69, 2.54.
10. Target descriptor, in this case the e-ink screen model number, which is ED060KC1.
11. Appended text, which is the numbers 0 through 9 repeated three times.

In this work, codes are referenced by shorthand notation of EC level and block size. For example, M30 refers to the level-M code with 30 pixels per block, with the following message: "M*4*33*1*30*1050*0.0845*83.7*2.54*ED060KC1*012345678901234567890123456789." As demonstrated in Fig. 1, many modern smartphones have native QR code scanners built into the camera software. The readers of this article can verify the QR code examples of M10 and M30 provided in Fig. 2 using a smartphone or similar device. As seen in Fig. 2, crosshairs are visible behind each code. These crosshairs were added for simulation analysis purposes only: they indicate the full width and height of the e-ink screen, placing the screen in context within the optical system's field of view.

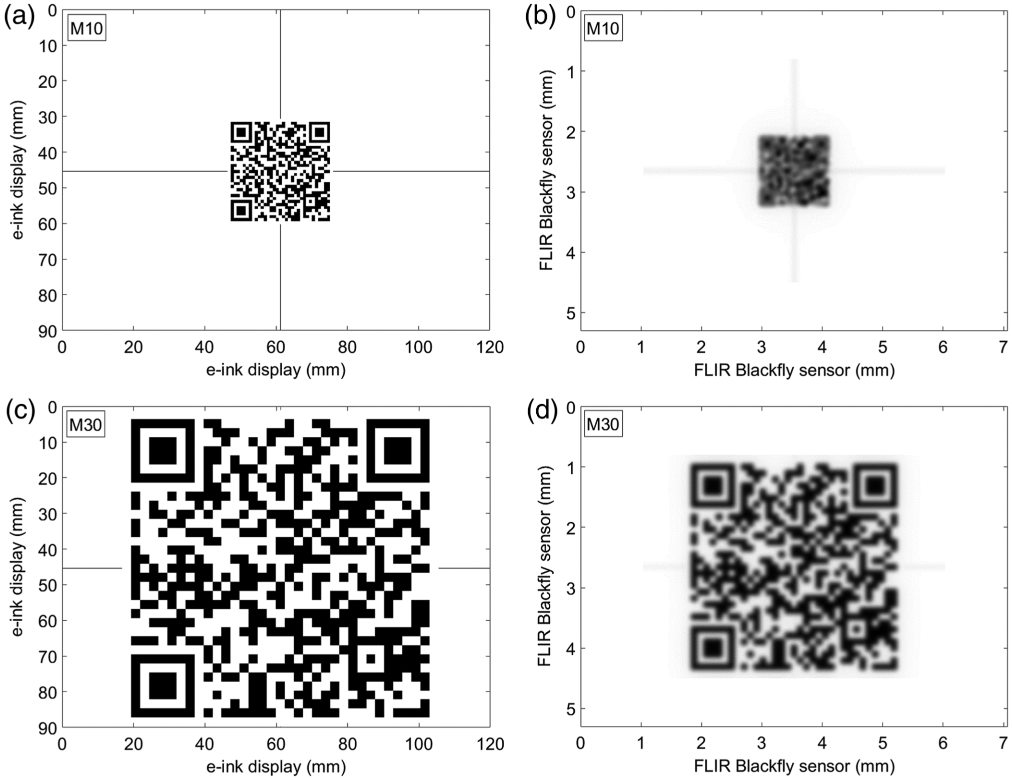


**Fig. 2** This example compares two e-ink screen display images (left column) with their degraded images (right column) after 45 m of simulated turbulent propagation with $C_n^2 = 8.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.01$ cm and subsequent imaging by the telescope and sensor. Panels (a) and (b) correspond to M10 codes (level-M EC and 10 e-ink pixels/block). While the (a) original M10 code is readable, the (b) degraded M10 code is unreadable. Panels (c) and (d) correspond to M30 codes (level-M EC and 30 e-ink pixels/block). The M30 code is readable in its (c) original and (d) degraded formats.

## 4 Feasibility Experiment

As an initial feasibility experiment, the five QR code images were uploaded to and cycled on the e-ink screen. These short-range (~0.5 m), qualitative results verified two important facts. First, images displayed correctly on the e-ink screen, with measured dimensions exactly matching the dimensions encoded in the images. Second, it was confirmed that QR code block size serves as an input for estimating optical resolution.

As seen in Fig. 1, QR codes were not detected when the constituent block size was below the optical system resolution. As soon as block size met or exceeded the minimum optical system resolution, code detection was successful. In this short-range experimental case, the M10 image displayed on the e-ink screen was the first code detected; as seen in Fig. 1(b), the block size is 0.85 mm. These qualitative tests verify concept feasibility, but such short-range bench tests do not incorporate atmospheric turbulence. The next sections focus on numerical simulations using MATLAB for quantitative analysis.

## 5 Turbulence Estimation Algorithm

### 5.1 Overview

The turbulence estimation procedure is comprised of the following steps.

1. Data collection of multiple spatially encoded target sizes to capture the transition point between successfully scanned codes and those that cannot be scanned.
2. Code information extraction to determine $\Delta x$ and QR code size at the transition point.
3. Automated range estimation using the QR code size.

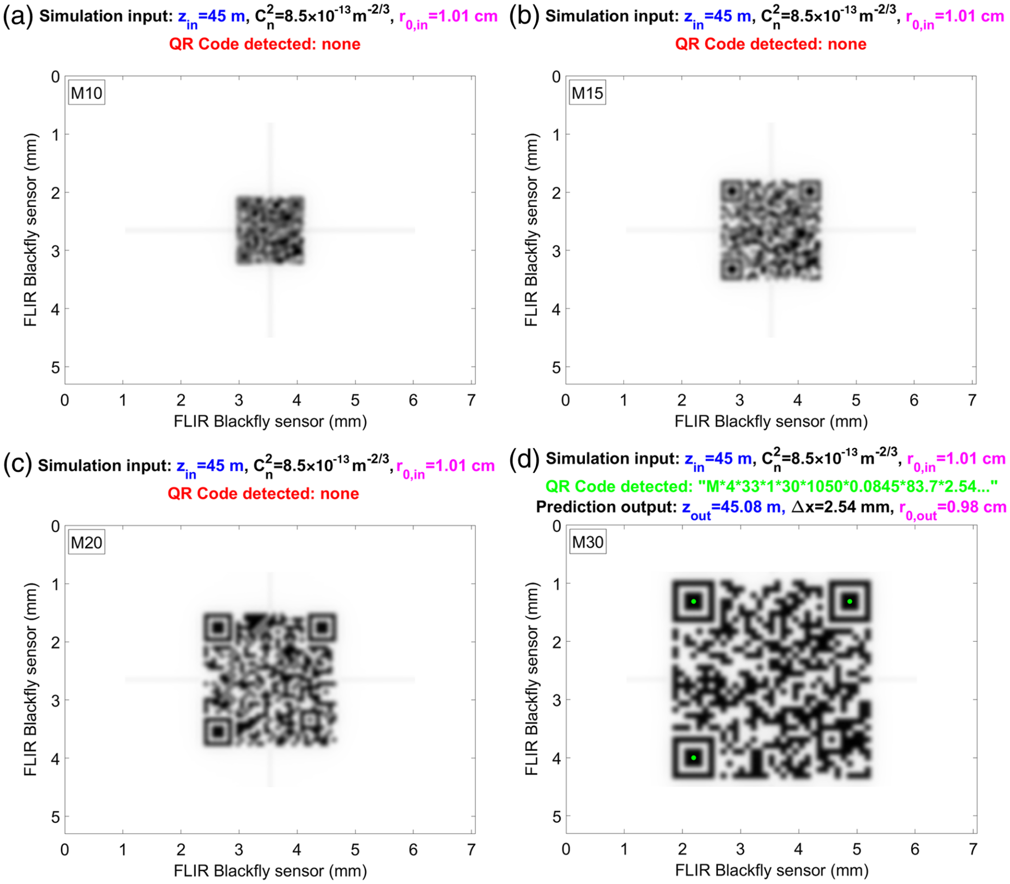
4. Finish with an output estimate of Fried's parameter, $r_{0,\mathrm{out}}$.

As described in step one, increasingly larger QR codes are cycled in time, and a transition point will indicate a shift from undetectable codes to successfully read codes. The code information shown in step two is extracted and saved for this smallest detectable QR code. The following section details steps three and four.

### 5.2 Automated Estimations of Range and Fried's Parameter

To extract information from the spatially encoded target, this work leveraged MATLAB's readBarcode.m function. In addition to reading the code information, the function outputs the spatial dimensions, enabling size assessment for range estimation. The function does not automatically output the full width of code (which, for this example, is 33 blocks), but instead outputs the finder pattern centroid locations (which, for this example, are each 26 blocks apart). Therefore, to properly compute $r_{0,\mathrm{est}}$, a centroid scale factor accounts for this difference.

From a practical perspective, three of the four QR code corners contain identical finder patterns; each pattern is seven by seven blocks in size, comprised of a (i) three by three black block in the center of the pattern surrounded by a (ii) white border one block wide and a (iii) black border one block wide. After the scanning algorithm detects the centers of the finder patterns, the next step is to compute the distance between the centroids. The QR code image size is the average of the first-to-second pattern distance and the second-to-third pattern distance. For the examples provided in this work, the image's pattern-to-pattern distance is 26 blocks in size and is referenced as $\mathrm{QR}_{i,26}$. The pattern centers are indicated by green dots in Fig. 3. These optional post-processing annotations aid visual verification for the user, and they are provided here for the


(a) Simulation input: $z_{in}$=45 m, $C_n^2$=8.5×$10^{-13}$ $m^{-2/3}$, $r_{0,in}$=1.01 cm
QR Code detected: none

(b) Simulation input: $z_{in}$=45 m, $C_n^2$=8.5×$10^{-13}$ $m^{-2/3}$, $r_{0,in}$=1.01 cm
QR Code detected: none

(c) Simulation input: $z_{in}$=45 m, $C_n^2$=8.5×$10^{-13}$ $m^{-2/3}$, $r_{0,in}$=1.01 cm
QR Code detected: none

(d) Simulation input: $z_{in}$=45 m, $C_n^2$=8.5×$10^{-13}$ $m^{-2/3}$, $r_{0,in}$=1.01 cm
QR Code detected: "M*4*33*1*30*1050*0.0845*83.7*2.54..."
Prediction output: $z_{out}$=45.08 m, Δx=2.54 mm, $r_{0,out}$=0.98 cm

**Fig. 3** These results show simulations of imaging through 45 m of atmosphere with input $r_{0,\mathrm{in}} = 1.01$ cm. Panels (a), (b), and (c) show the unreadable M10, M15, and M20 codes, respectively. Panel (d) shows detection starting at M30. Range is estimated at 45.08 m. Spatial resolution is estimated at 2.54 mm and output $r_{0,\mathrm{out}} = 0.98$ cm.

reader, but algorithmically they inhibit further code detection by basic scanning software routines.

As described in Sec. 3.2, each QR code data string provides the physical size of the code, from black edge to black edge, in units of mm, and corresponds to 33 blocks for these examples. Here, the object's edge-to-edge distance is 33 blocks in size and is referenced as $\mathrm{QR}_{o,33}$. To properly relate object size to image size, the physical QR code object size ($\mathrm{QR}_{o,33}$) interpreted by readBarcode.m is scaled down by centroidScale = 26 blocks/33 blocks = 0.788. Through this scaling, QR code object size is accurately represented. Recalling the theoretical approach above [Eq. (6)], range is then given $z = f_l \times \mathrm{QR}_{o,33} \times \text{centroidScale}/\mathrm{QR}_{i,26}$.

With propagation range restated in terms of practical measurements and specifications, Eq. (7) for estimated $r_{0,\text{est}}$ is reframed as the following output:

$$r_{0,\text{out}} = \lambda \frac{f_l \times QR_{o,33} \times \text{centroidScale}}{QR_{i,26}} \frac{1}{\Delta x}. \tag{8}$$

At this point, the mathematical theory and estimation algorithm are now complete. Within MATLAB, the next steps are to apply the turbulence simulation blur kernels, scan these results with the estimation algorithms, and automatically estimate Fried's coherence parameter, $r_{0,\text{out}}$.

# 6 Quantitative Simulations

## 6.1 Results

For this initial work, two ranges were selected for simulations: 45 m and 90 m. These moderate ranges allow for the study of intermediate to severe turbulence while also facilitating a convenient image sampling. Recall that the object plane's e-ink pixel size is $Px_o = 0.0845$ mm, the image plane's camera pixel size is $Px_i = 3.45$ $\mu$m, and the focal length is $f_l = 1836$ mm. The sampling ratio of object plane pixels to the image plane pixels is expressed by pixel fill factor $F = ((f_l/Px_i)/(z/Px_o))^2$. In general, this type of mapping is useful to consider when simulating the sampling of display screens with imaging sensors, but it also finds utility in similar applications, including co-registered mapping of 3D LIDAR to 2D imaging sensors.[24]

Conveniently, when range $z = 45$ m, $F \approx 1$; in this case, one e-ink pixel is sampled by one sensor pixel. This results in a 1:1 image remapping, but as seen in Figs. 2(a) and 2(b), the scale appears slightly different. This small difference arises from the sensor's increased pixel count: the camera's sensor has 2048 pixels by 1536 pixels while the e-ink screen has 1448 pixels by 1072 pixels. Therefore, the camera has a field of view that is slightly larger than the e-ink screen, which is seen as white space surrounding the crosshair pattern. When range is $z = 90$ m, $F \approx 0.25$; in this case, one e-ink pixel is sampled within one fourth of the sensor's camera pixel. Restated, each camera pixel samples four e-ink screen pixels, and image resolution is reduced at this more distant range of 90 m.

As an introductory example, Fig. 2 highlights the capability differences between smaller (M10) and larger (M30) QR codes with and without severe turbulence ($C_n^2 = 8.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.01$ cm as simulation inputs). Panels (a) and (b) show the M10 code before and after 45 m propagation: the former is readable, but the latter is not. However, panels (c) and (d) show that the M30 code remains readable both before and after 45 m propagation. These results can be verified by the reader using a modern smartphone QR code scanner.

The images of Fig. 3 illustrate the procedure of cycling QR code sizes for a given range and turbulence. The title of each panel provides simulation inputs and the results from the estimation software. Of particular note is the green text annotating the extracted information for successfully scanned codes. This example shows four of the five QR code sizes; here, the input range is 45 m, $C_n^2 = 8.5 \times 10^{-13}$ m$^{-2/3}$, and $r_{0,\text{in}} = 1.01$ cm. When reading sequentially from the smallest code to the largest code, the first successful detection indicates the spatial resolution ($\Delta x$) of the optical system. Figs. 3(a)–3(c) show the unreadable M10, M15, and M20 codes, respectively. As seen in Fig. 3(d) and plotted in Fig. 4(d), detection occurs at M30. Through the relation described in Eq. (6), range is estimated at 45.08 m, which is nearly equal to the 45 m input range. Spatial

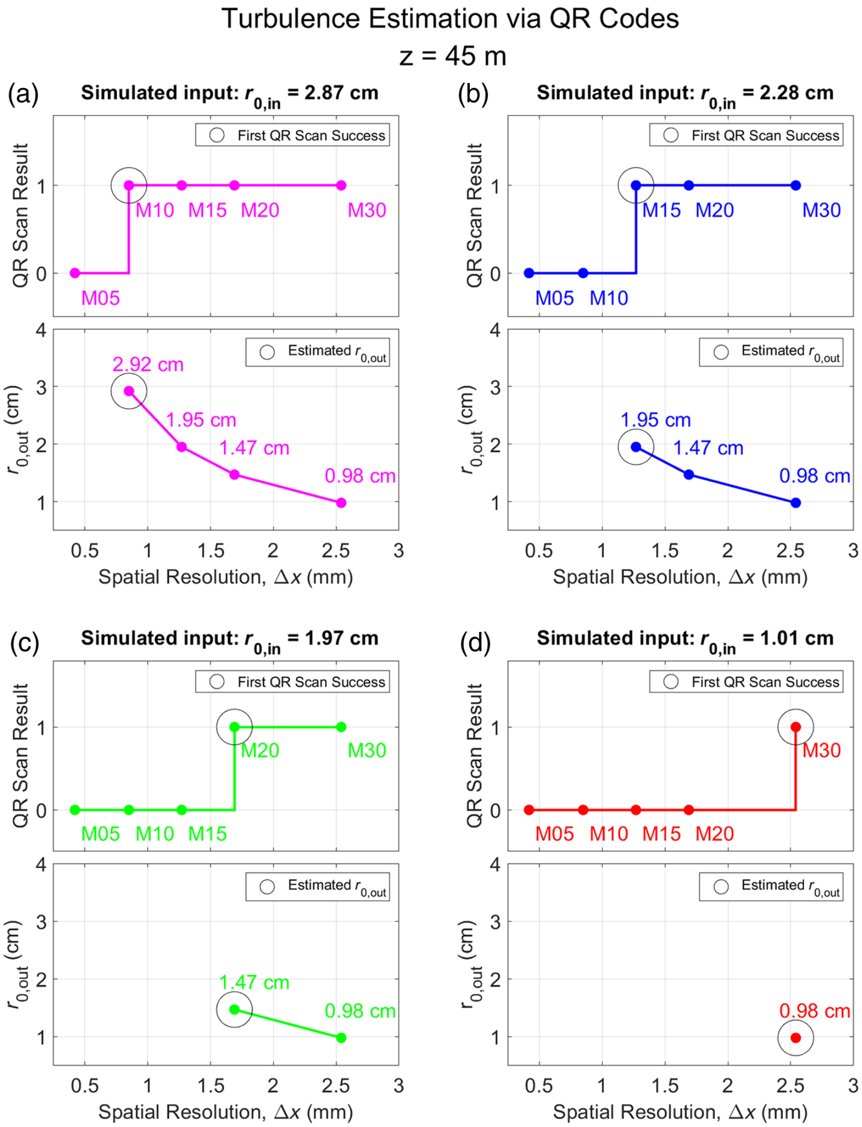


**Fig. 4** QR code imaging simulations at 45 m are shown. Four levels of turbulence are explored. Panel (a) inputs are $C_n^2 = 1.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 2.87$ cm; detection starts at M10 with $r_{0,\text{out}} = 2.92$ cm. Panel (b) inputs are $C_n^2 = 2.2 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 2.28$ cm; detection starts at M15 with $r_{0,\text{out}} = 1.95$ cm. Panel (c) inputs are $C_n^2 = 2.8 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.97$ cm; detection starts at M20 with $r_{0,\text{out}} = 1.47$ cm. Panel (d) inputs are $C_n^2 = 8.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.01$ cm; detection starts at M30 with $r_{0,\text{out}} = 0.98$ cm.

resolution (i.e., QR code block size) is estimated at 2.54 mm. Following Eq. 7, $r_{0,\text{out}}$ is estimated at 0.98 cm.

Figure 4 provides full, compiled results from 45 m propagation simulations using four turbulence levels to better illustrate the concept. Panel (a) shows simulations using input values of $C_n^2 = 1.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 2.87$ cm. As seen in upper panel (a), QR code detection starts at M10. Lower panel (a) shows calculated $r_{0,\text{out}} = 2.92$ cm for this first successful detection, which is 0.05 cm above the input value. Subsequent detections of larger codes are ignored. Panel (b) shows results using inputs of $C_n^2 = 2.2 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 2.28$ cm. Detection starts at M15, with estimated $r_{0,\text{out}} = 1.95$ cm, which is 0.33 cm below the input value. Panel (c) shows results using inputs of $C_n^2 = 2.8 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.97$ cm. Detection starts at M20, with estimated $r_{0,\text{out}} = 1.47$ cm, which is 0.50 cm below the input value. Panel (d) shows results using inputs of $C_n^2 = 8.5 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\text{in}} = 1.01$ cm. Detection starts at M30, with estimated $r_{0,\text{out}} = 0.98$ cm, which is 0.03 cm below the input value. Overall, 45 m

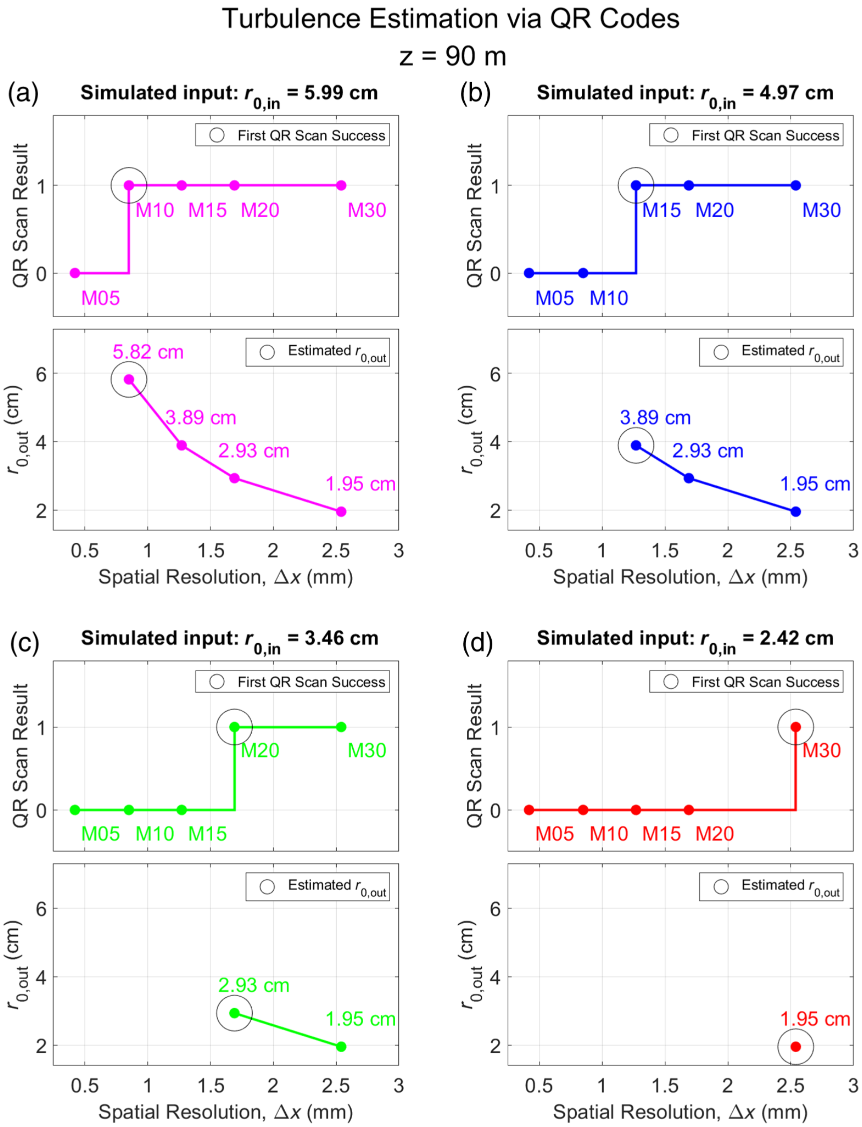


**Fig. 5** QR code imaging simulations at 90 m are shown. Four levels of turbulence are explored. Panel (a) inputs are $C_n^2 = 2.2 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 5.99$ cm; detection starts at M10 with $r_{0,\mathrm{out}} = 5.82$ cm. Panel (b) inputs are $C_n^2 = 3.0 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 4.97$ cm; detection starts at M15 with $r_{0,\mathrm{out}} = 3.89$ cm. Panel (c) inputs are $C_n^2 = 5.5 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 3.46$ cm; detection starts at M20 with $r_{0,\mathrm{out}} = 2.93$ cm. Panel (d) inputs are $C_n^2 = 1.0 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 2.42$ cm; detection starts at M30 with $r_{0,\mathrm{out}} = 1.95$ cm.

simulation accuracy is favorable, with discrepancies ranging from −0.50 cm to +0.05 cm and averaging to a value of −0.20 cm.

Figure 5 provides the full, compiled results from 90 m propagation simulations. Panel (a) shows simulations using input values of $C_n^2 = 2.2 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 5.99$ cm. As seen in upper the panel (a), QR code detection starts at M10. Lower panel (a) shows calculated $r_{0,\mathrm{out}} = 5.82$ cm for this first successful detection, which is 0.17 cm below the input value. Panel (b) shows results using inputs of $C_n^2 = 3.0 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 4.97$ cm. Detection starts at M15, with estimated $r_{0,\mathrm{out}} = 3.89$ cm, which is 1.08 cm below the input value. Panel (c) shows results using inputs of $\mathrm{C}_n^2 = 5.5 \times 10^{-14}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 3.46$ cm. Detection starts at M20, with estimated $r_{0,\mathrm{out}} = 2.93$ cm, which is 0.53 cm below the input value. Panel (d) results have inputs of $C_n^2 = 1.0 \times 10^{-13}$ m$^{-2/3}$ and $r_{0,\mathrm{in}} = 2.42$ cm. Detection starts at M30, with estimated $r_{0,\mathrm{out}} = 1.95$ cm, which is 0.47 cm below the input value. Overall, 90 m simulation accuracy is favorable, with discrepancy values that range from −1.08 cm to −0.17 cm and average to a value of –0.56 cm.

### 6.2 Discussion

The primary metric by which imaging systems are judged is their spatial resolution, $\Delta x$, which defines the minimum separation distance between objects or features that can be resolved by the optical system. Under perfect conditions, the optical sensor and scanner algorithm will work in concert to detect QR codes with block sizes that match the spatial resolution. In practice, sampling by the finite optical sensor and scanning by the reader algorithms require block sizes slightly larger than the theoretical spatial resolution to achieve detection. Thus, estimated $\Delta x$ values slightly exceed most theoretical predictions, which, as detailed above, generally result in smaller $r_{0,\mathrm{out}}$ values than anticipated. Nonetheless, QR code block size serves as a realistic estimate for $\Delta x$.

The simulation results presented in this work span a limited number of turbulence levels. Future work will explore additional turbulence levels to improve the accuracy of turbulence estimates, as $r_{0,\mathrm{out}}$ depends on the threshold of QR code detection. In an experimental scenario, turbulence is an uncontrolled factor; by reducing the step size and increasing the number of QR codes used, a more precise turbulence estimation threshold will produce more accurate $r_{0,\mathrm{out}}$ estimates.

## 7 Summary

Through a series of preliminary experiments and advanced image simulations using spatially encoded QR codes, the concept of remotely estimating optical turbulence has been verified. Knowledge of the propagation range and QR code size is not required ahead of time. Each QR code contains spatially encoded information detailing its constituent block size and overall physical size, enabling automated calculations of spatial resolution $\Delta x$ and range distance $z$, respectively. From these two values ($\Delta x$ and $z$), algorithms estimate the optical turbulence strength, expressed as Fried's parameter, $r_0$. To physically realize this concept, the spatially encoded targets are cycled in time on a low-power e-ink screen, enabling automated estimates of optical turbulence.

Future theoretical work will involve modulation transfer function and line spread function analysis[25] to further compare this QR code-based method to established optical measurement techniques.

Future experimental work will involve creating and printing a series QR codes on outdoor vinyl banners for optical tests using a standard imaging camera in conjunction with known consumer off-the-shelf scintillometers and sensors that derive estimates of $C_n^2$. Current banner designs include an extensive range of QR code sizes, from 15 mm wide to 1219 mm wide. The ground-truth turbulence measurement devices under current consideration are the Scintec BLS, the MZA Delta, the Blue Halo Optical Differential Temperature Sensor, and traditional nodal $C_n^2$ measurements derived from sonic anemometers. Conducting such experiments will advance the theory from the current feasibility stage to a validated remote turbulence estimation method.

### Disclosures

The authors have no conflicts of interest to disclose.

### Code and Data Availability

QR codes were generated using the procedure described in Sec. 3.2 and can be freely recreated by the reader. Modeling and simulation code routines and additional data are available from the authors upon request, and with permission from NIWC Pacific.

### Acknowledgments

The authors would like to thank the Naval Innovative Science and Engineering Fund for its financial support of this research.

## References

1. D. L. Fried, "Optical resolution through a randomly inhomogeneous medium for very long and very short exposures," *J. Opt. Soc. Am.* **56**, 1372–1379 (1966).
2. A. N. Kolmogorov, "The local structure of turbulence in incompressible viscous fluids for very large Reynolds numbers," in *Turbulence, Classic Papers on Statistical Theory*, S. K. Friedlander, Ed., Wiley-Interscience, New York (1961).
3. V. I. Tatarskii, *Wave Propagation in a Turbulent Medium*, 1st ed., Dover Publications, New York (1967).
4. A. Ishimaru, *Wave Propagation and Scattering in Random Media*, 1st ed., Vol. 2, Academic Press, New York (1978).
5. J. W. Goodman, *Statistical Optics*, 1st ed., John Wiley & Sons, New York (1985).
6. M. Roggemann and B. Welsh, *Imaging Through Turbulence*, 1st ed., CRC Press, Boca Raton, Florida (1996).
7. C.-M. Gan et al., "Application of active optical sensors to probe the vertical structure of the urban boundary layer and assess anomalies in air quality model PM2.5 forecasts," *Proc. SPIE* **8182**, 81820P (2011).
8. M. Huang, B. Huang, and A. H.-L. Huang, "Application of Intel many integrated core (MIC) architecture to the Yonsei University planetary boundary layer scheme in Weather Research and Forecasting model," *Proc. SPIE* **9247**, 92470J (2014).
9. D. Estrada et al., "Multi-frame image fusion using a machine learning-based weight mask predictor for turbulence-induced image degradation," *J. Appl. Remote. Sens.* **17**(1), 016514 (2023).
10. N. Ferlic, A. Laux, and L. Mullen, "Radio frequency intensity modulation vs. heterodyne modulation on laser beams traversing a random optical phase distortion," *Proc. SPIE* **12543**, 1254302 (2023).
11. L. J. Angot, C.-C. Chang, and Y.-L. Chen, "Evaluating distances using a coded lens camera and a blur metric," *Proc. SPIE* **7877**, 78770O (2011).
12. F. Liu, J. Wu, and L. Cao, "Autofocusing of Fresnel zone aperture lensless imaging for QR code recognition," *Opt. Express* **31**(10), 15889–15903 (2023).
13. Q. Wu and Y. He, "Indoor location technology based on LED visible light and QR code," *Appl. Opt.* **60**(16), 4606–4612 (2021).
14. G. Bolanakis, K. Nanos, and E. Papadopoulos, "A QR Code-based high-precision docking system for mobile robots exhibiting submillimeter accuracy," in *IEEE/ASME Int. Conf. Adv. Intell. Mechatron. (AIM)*, Delft, Netherlands, pp. 830–835 (2021).
15. E. Olson, "AprilTag: a robust and flexible visual fiducial system," in *IEEE Int. Conf. Rob. and Autom.*, Shanghai, China, pp. 3400–3407 (2011).
16. K. R. Drexler, S. D. Lilledahl, and B. Laxton, "Digital adaptive optics for turbulence mitigation," *Proc. SPIE* **11834**, 118340D (2021).
17. K. R. Drexler, S. D. Lilledahl, and B. Laxton, "Atmospheric mitigation for imaging applications," *Proc. SPIE* **12118**, 121180J (2022).
18. S. Lilledahl et al., "Digital adaptive optics for turbulence mitigation with QR codes," *Proc. SPIE* **12237**, 122370D (2022).
19. B. Neuner et al., "Digital adaptive optics with interferometric homodyne encoding for mitigating atmospheric turbulence," *Opt. Eng.* **62**(2), 023104 (2023).
20. B. Neuner et al., "Digital adaptive optical imaging for oceanic turbulence mitigation," in *IEEE Underwater Technol. (UT)*, Tokyo, Japan, pp. 1–5 (2023).
21. L. C. Andrews, *Field Guide to Atmospheric Optics*, 2nd ed., Vol. FG41, p. 75, SPIE Press, Bellingham, Washington (2019).
22. K. Erkan, "QR code and 2D code generator," https://keremerkan.net/qr-code-and-2d-code-generator/ (accessed 11 April 2023).
23. N. Berry, "Wounded QR codes," https://datagenetics.com/blog/november12013/index.html (accessed 27 June 2022).
24. T. Foster, A. Dalal, and J. Ball, "RGB pixel-block point-cloud fusion for object detection," *Proc. SPIE* **11748**, 1174804 (2021).
25. F. Viallefont-Robinet et al., "Comparison of MTF measurements using edge method: towards reference data set," *Opt. Express* **26**(26), 33625–33648 (2018).

**Burton Neuner III** received his BS degree in engineering physics from the University of Illinois at Urbana-Champaign in 2004 and his PhD in physics from the University of Texas at Austin in 2011. He is a research scientist at NIWC Pacific, San Diego, California, United States. His current research interests include oceanic and atmospheric optical turbulence, optical communication, environmental sensing, and machine learning.

**Skylar D. Lilledahl** received his BS degree in physics from San Diego State University in 2019, focusing on modern optical communications. He is a research scientist at NIWC Pacific, San Diego, California, United States, focusing on optical atmospheric propagation. His current research is in mitigating atmospheric optical turbulence in imaging and communications systems.

**Kyle R. Drexler** received his PhD in electrical engineering from Michigan Technological University in 2012 with a focus on phase compensation algorithms for mitigating atmospheric turbulence in free space optical communications. He is a senior research scientist at NIWC Pacific, San Diego, California, United States, and is currently developing electro-optical propagation and atmospheric turbulence estimation techniques for imaging and directed energy applications. Since 2007, he has built system models, optical simulations, reconstruction algorithms, and hardware validation routines.