Blind Deblurring of Barcodes via Kullback-Leibler Divergence

Gabriel Rioux, Christopher Scarvelis, Rustum Choksi, Tim Hoheisel, and Pierre Maréchal

Abstract—Barcode encoding schemes impose symbolic constraints which fix certain segments of the image. We present, implement, and assess a method for blind deblurring and denoising based entirely on Kullback-Leibler divergence. The method is designed to incorporate and exploit the full strength of barcode symbologies. Via both standard barcode reading software and smartphone apps, we demonstrate the remarkable ability of our method to blindly recover simulated images of highly blurred and noisy barcodes. As proof of concept, we present one application on a real-life out of focus camera image.

Index Terms—Blind deblurring, denoising, symbology, QR barcode, UPC barcode, maximum entropy on the mean, Kullback-Leibler divergence, Fenchel-Rockafellar duality, L-BFGS

1 Introduction

Deblurring in image processing addresses a notoriously difficult ill-posed problem. In this article we present a novel algorithm for deblurring and denoising of barcodes. The strength of our method lies in its effective incorporation (at all stages) of the precise symbology of barcodes. In principle, our method could apply to any class of images possessing some a priori set structure. We present and test the method for barcodes for the following reasons: (i) Barcodes remain ubiquitous objects for the encoding of information, and are the simplest class of images which follow a fixed symbology. (ii) For large amounts of blurring and noise, there is a less ambiguous test of the success of the algorithm than the *eye norm* – their readability by standard commercial software and smartphone *apps*.

One dimensional (1D) UPC barcodes remain popular for coding merchandise while *Quick Response* (QR) barcodes, a type of matrix 2D barcode [37], [42], are increasingly popular because of the ubiquity of smartphone cameras. While barcode readers and smartphone apps are well-developed, the issue of deblurring and denoising barcodes remains of considerable interest with the presence of motion blur from hand movement and noise intrinsic to the camera sensor. The interplay between deblurring and barcode symbology is important for the successful use of mobile smartphones [8], [12], [24], [50], [51], [53]. Methods for deblurring and denoising of barcode signals are well-developed; for example, many techniques have been presented in academic articles (see, for example, [4], [5], [10], [13], [16], [18], [21], [22], [25], [27], [28], [31], [32], [33], [39], [41], [49], [51], [52]) while implemented algorithms are hidden in commercial software (for example, open source readers like *Zbar* and apps like Apple's *QR Reader*).

The majority of general state-of-the-art blind deblurring methods approach the problem in two steps. The first step is to estimate the blurring kernel and the second is to use non-blind deblurring methods to estimate the original image using the estimated kernel (cf. [7], Chapter 1 of [43] and the references therein, [26]). Our approach follows this structure, however we present novel kernel estimation and deblurring methods that are based on an approach known as the *Method of Maximum Entropy on the Mean* (MMEM) through the Kullback-Leibler divergence. To this end, we do not attempt to find the cleaned image directly but rather we find its probability density function over all binary arrays. We then take, as our best guess of the cleaned image, its (thresholded) expectation. While this particular use of entropy and the Kullback-Leibler divergence has a well-established record of success in many areas of information theory (cf. [1], [11]), we believe this is the first implementation for deblurring of barcodes. In fact, while the Kullback-Leibler divergence appears in the highly-cited deblurring paper of Fergus et al. [14], to our knowledge this particular approach is also new within the wider context of image deblurring. As can be seen in Figs. 7, 8, 9, and 10, our method is quite remarkable in its ability to blindly deblur and denoise data. In each case, the only information used to reconstruct the barcode from the simulated blurred and noisy signal is the QR symbology (cf. Fig. 1). Software (Zbar and smartphones) were all unable to read the initial signal; however, all can read our processed versions. To our knowledge, we are unaware of any other simple method which can produce such dramatic results.

The principle of maximum entropy was introduced by E.T. Jaynes in 1957 [19], [20]. This principle states that among all

- *G. Rioux, C. Scarvelis, R. Choksi, and T. Hoheisel are with the Department of Mathematics and Statistics, McGill University, Montréal, QC H3A 0G4, Canada. E-mail: {gabriel.rioux, christopher.scarvelis}@mail.mcgill.ca, {rustum.choksi, tim.hoheisel}@mcgill.ca.*
- *P. Maréchal is with the Institut de Mathématiques de Toulouse, Université Paul Sabatier, Toulouse 31 062, France. E-mail: pierre.marechal@math.univ-toulouse.fr.*

Manuscript received 21 Dec. 2018; revised 13 May 2019; accepted 1 July 2019. Date of publication 9 July 2019; date of current version 3 Dec. 2020. (Corresponding author: Rustum Choksi.) Recommended for acceptance by E. Shechtman. Digital Object Identifier no. 10.1109/TPAMI.2019.2927311

0162-8828 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

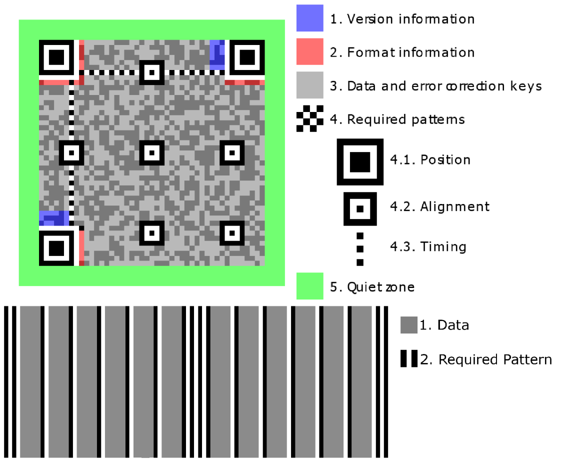

Fig. 1. A depiction of the symbolic constraints in UPC-A and QR codes (Source (top image): Wikipedia [42] (image by Bobmath, CC BY-SA 3.0).

probability distributions that are compatible with given moments, the least biased is the one that maximizes the entropy. If prior knowledge on the unknown distribution is available, then the Kullback-Leibler relative entropy is the method of choice. A particular occurrence of it is named the MMEM which was introduced by Dacunha-Castelle and Gamboa [11], and implemented later on in various applications (see e.g., [1], [34]). Applying the MMEM entails solving a convex program in possibly infinite dimensions under finitely many affine equality constraints. This type of problem is efficiently approached by means of Fenchel-Rockafellar duality [2], [3], [35], [44]. In our application we consider a finite dimensional problem; albeit one of "very high" dimension.

We briefly outline our entropic barcode method; the details of the algorithm are presented in Section 2.

1.1 Outline of Our Kullback-Leibler Approach

We model a barcode by a vector $\boldsymbol{x} \in \{0,1\}^N$ of N independent Bernoulli random variables with x_i denoting the ith bar for a UPC barcode or the ith module for a QR code. For UPC barcodes $N = 95$ while for QR barcodes N ranges from 441 to 31329. We model the blurring of the barcode $\boldsymbol{x}$ via discrete linear convolution of the form $\boldsymbol{b} = C\boldsymbol{x} \equiv \boldsymbol{c} * U\boldsymbol{x}$, where $C \in \mathbb{R}^{Nm\times N}$, m is an *upscaling factor* as explained at the start of Section 2 and $U \in \mathbb{R}^{Nm\times N}$ is the matrix that upscales $\boldsymbol{x}$. C is therefore responsible for upscaling and blurring $\boldsymbol{x}$ with *point spread function* (PSF) $\boldsymbol{c} \in \mathbb{R}^{Nm}$, also known as the blur kernel, and $\boldsymbol{b} \in \mathbb{R}^{Nm}$ the observed blurry signal. Let us for the moment assume that C is known (this is the case for non-blind deblurring). The number of possible barcodes is 2^N and we let $\boldsymbol{p}$ be a *probability mass function* (PMF) defined over the space of barcodes $\{0,1\}^N$. Hence, $\boldsymbol{p} \in \mathbb{R}^{2^N}$, where the kth component of $\boldsymbol{p}$, denoted p_k (this notation will be used throughout) represents the probability assigned to the kth barcode in $\{0,1\}^N$. Now, given a prior distribution μ over the space of barcodes, we minimize the function

$$p \mapsto \sum_{i=1}^{2^N} p_i \log\left(\frac{p_i}{\mu_i}\right) + \gamma \left\| C\mathbb{E}_{\boldsymbol{p}}[\boldsymbol{x}] - \boldsymbol{b} \right\|^2, \tag{1}$$

over all PMFs $\boldsymbol{p}$. With the solution $\bar{\boldsymbol{p}}$ in hand, our cleaned (processed) barcode is then the thresholded expectation of $\bar{\boldsymbol{p}}$, $\mathbb{E}_{\bar{\boldsymbol{p}}}[\boldsymbol{x}]$. Ideally $\bar{\boldsymbol{p}}$ should be a 1-hot vector such that it gives full weight to a single barcode. We consider a uniform prior, a prior based entirely on the symbology (i.e., one which assigns probability 0 to any $\boldsymbol{x}$ which does not respect the symbology), and for UPC-A barcodes an empirically generated prior based upon a database of 10^6 barcodes.

Even with C known, problem (1) with our range of N, is numerically intractable. To this end, we employ the following strategy. First, we exploit Fenchel-Rockafellar duality with a significantly simplified dual problem which has Nm degrees of freedom as opposed to 2^N for the primal problem (1). While this presents a fundamental reduction in complexity, it is still too costly to compute $\bar{\boldsymbol{p}}$ via the solution to the dual problem. On the other hand, we do not need to find $\bar{\boldsymbol{p}}$ but rather its expectation, and to this end we present a probabilistic version of the dual which allows for the quick and efficient computation of $\bar{\boldsymbol{x}} = \mathbb{E}_{\bar{\boldsymbol{p}}}[\boldsymbol{x}]$.

The above outlines the method when C is known. For blind deblurring, i.e., when C is unknown, we perform an iterative process which couples the above with an entropy based optimization (cf. (5) in the following section) to estimate $\boldsymbol{c}$ from the observed signal $\boldsymbol{b}$ and $\bar{\boldsymbol{x}}$, where $\bar{\boldsymbol{x}}$ is the outcome of the previous entropic image estimation. The iteration begins with an initial estimation of $\boldsymbol{c}$ based upon $\boldsymbol{b}$ and $\bar{\boldsymbol{x}} = \mathbb{E}_{\mu}[\boldsymbol{x}]$.

2 The Entropic Blind Deblurring Method

Throughout, the process of capturing an image will be modeled via $\boldsymbol{b} = \boldsymbol{c} * U\boldsymbol{x} = C\boldsymbol{x}$ where $\boldsymbol{x} \in \{0,1\}^N$ is the original barcode, $C \in \mathbb{R}^{Nm\times N}$ is a matrix that upsamples and blurs the image via discrete linear convolution by the PSF $\boldsymbol{c}$ and $\boldsymbol{b} \in \mathbb{R}^{Nm}$ is the acquired image. We model the unknown true barcode $\boldsymbol{x}$ as a vector $\boldsymbol{X} = (X_1, \ldots, X_N)$ of N independent Bernoulli random variables and recall $N \in \mathbb{N}$ is the total number of the barcode modules. We let $m \in \mathbb{N}$ be an upscaling factor, as the pixels of a camera will seldom align in a one-to-one manner with the bars of the barcode. For example, if $m = 3$, one module of a QR code will correspond to a block of 3×3 pixels rather than just one pixel. Moreover, upscaling is necessary in our model to consider realistic quantities of blurring as demonstrated in Fig. 4.

We represent the probability mass function as a vector $\boldsymbol{p} \in \Delta_{2^N}$, where the ith component of $\boldsymbol{p}$ corresponds to the probability $p(x^i)$ of the ith binary sequence in $\{0,1\}^N$ under some arbitrary ordering of the set. We use the symbol Δ_n to denote the unit simplex in $\mathbb{R}^n$ defined as

$$\Delta_n = \left\{ \boldsymbol{u} \in \mathbb{R}^n \;:\; \sum_{i=1}^{n} u_i = 1,\; u_i \geq 0 \,(i = 1, \ldots, n) \right\}.$$

The unit simplex Δ_n is the space of probability distributions over a finite sample space of cardinality n.

The Kullback-Leibler relative entropy quantifies the divergence between two probability distributions and is defined in [29, Eqn. 2.4] as

$$\mathscr{K}(\boldsymbol{p};\mu) = \begin{cases} \sum_{i\in I} p_i \log\left(\frac{p_i}{\mu_i}\right) & \text{for } \boldsymbol{p} \in \Delta_{2^n} \\ +\infty & \text{otherwise} \end{cases},$$

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

where $I = \{j : \mu_j > 0\}$. Working in the convention that $0\log 0 = 0$ and that $p_k = 0$ for some k if and only if $\mu_k = 0$, summing over I is equivalent to summing from $i = 1, \ldots, 2^N$, as if $\mu_j = 0$ for some j, the jth summand is 0. These constraints ensure that the entropy term is well-defined. Moreover, μ denotes a prior probability distribution which encodes certain characteristics which a valid barcode should exhibit.

The constraint on the mean can be rephrased by noting that $\mathbb{E}_{\boldsymbol{p}}(\boldsymbol{x}) = A\boldsymbol{p}$ where $A \in \{0,1\}^{N\times 2^N}$ is a matrix formed by ordering the set of all binary sequences of length N and letting the ith column of A be the ith element of this ordering. Thus, A computes the expectation value. This constraint will be enforced by means of the penalty function

$$\boldsymbol{p} \mapsto \gamma\|\boldsymbol{c} * UA\boldsymbol{p} - \boldsymbol{b}\|^2. \tag{2}$$

Here, $\gamma > 0$ is a scalar which can be varied in order to penalize deviations from the mean to a variable extent. We make precise that the standard euclidean norm will be used throughout. This form of penalization is flexible enough to permit the presence of additive noise in the image acquisition process without needing to explicitly account for it. Indeed, with noise, the observed barcode $\boldsymbol{b}$ will not generally be equal to $\boldsymbol{c} * UA\boldsymbol{p}$ for any $\boldsymbol{p}$. Hence, a hard constraint on the mean enforcing that $\boldsymbol{c} * UA\boldsymbol{p}$ must equal $\boldsymbol{b}$ is inadequate.

In the case of blind deblurring, we seek to determine the PMF $\bar{\boldsymbol{p}}$ and the convolution kernel $\bar{\boldsymbol{c}}$ which solve

$$\inf_{\boldsymbol{p},\boldsymbol{c}}\Big\{\mathscr{K}(\boldsymbol{p};\mu) + \mathscr{K}(\boldsymbol{c};\nu) + \gamma\|\boldsymbol{c} * UA\boldsymbol{p} - \boldsymbol{b}\|^2\Big\}, \tag{3}$$

as $\bar{\boldsymbol{p}}$ will allow us to estimate the original barcode and the PSF responsible for the blurring is unknown. In this equation, μ and ν are distinct prior probability distributions and the particular characteristics that μ and ν encode will be discussed in the following sections, as they play a fundamentally different role. In this framework, the Kullback-Leibler divergence also guarantees that $\bar{\boldsymbol{p}}$, $\bar{\boldsymbol{c}}$ are elements of the 2^N-simplex by its very definition. The utility of this property will be made clear later. Our approach to tackling problem (3) is by iteratively coupling the following subproblems.

1. Image estimation based on $\boldsymbol{c}$ (non-blind deblurring): Determine $\bar{\boldsymbol{p}}$ as a solution of

$$\inf_{\boldsymbol{p}}\Big\{\mathscr{K}(\boldsymbol{p};\mu) + \tfrac{\alpha}{2}\|\boldsymbol{c} * UA\boldsymbol{p} - \boldsymbol{b}\|^2\Big\}. \tag{4}$$

   Here, based on the PSF $\boldsymbol{c}$ we determine an approximation of the image through $A\bar{\boldsymbol{p}}$.
2. Kernel estimation based on $\boldsymbol{p}$: Determine $\bar{\boldsymbol{c}}$ as a solution of

$$\inf_{\boldsymbol{c}}\Big\{\mathscr{K}(\boldsymbol{c};\nu) + \tfrac{\beta}{2}\|\boldsymbol{c} * UA\boldsymbol{p} - \boldsymbol{b}\|^2\Big\}. \tag{5}$$

   Here, based on the image $A\boldsymbol{p}$ we approximate the PSF $\bar{\boldsymbol{c}}$.

Alternating between image and kernel estimation is common in state of the art deblurring methods (see e.g., [6], [38]). In the following section we first discuss how to solve the problems (4) and (5), respectively, and then we discuss the coupling mechanism which constitutes the basis for our algorithm.

2.1 The Image Estimation

Throughout this section, (4) will be referred to as the primal problem. Recalling that both the convolution and expectation operators can be written in matrix form, we define $M = CA$ with $M \in \mathbb{R}^{Nm\times 2^N}$ for the sake of convenience. We note, moreover that solving this problem is not a straightforward endeavour, as it is a 2^N-dimensional minimization problem. Even in the simpler case of UPC-A encoding, a barcode is composed of 95 bars, hence $\boldsymbol{p} \in \mathbb{R}^{2^{95}}$. In such a high dimensional minimization problem, attempting to compute a solution directly is infeasible and thus an alternative method must be determined to solve (4).

2.1.1 A Convex Analytic Approach to Solving the Primal Problem

We employ Fenchel-Rockafellar duality for a first simplification of the problem (4). To this end, we present a brief exposition of this duality scheme following [45, Example 11.41]: For $\phi : \mathbb{R}^\ell \to \mathbb{R}\cup\{+\infty\}$ its domain is $\operatorname{dom}\phi := \{\boldsymbol{x}\in\mathbb{R}^\ell \,|\, \phi(\boldsymbol{x}) < +\infty\}$. Its conjugate $\phi^* : \mathbb{R}^\ell \to \mathbb{R}\cup\{\pm\infty\}$ is given by $\phi^*(\boldsymbol{y}) = \sup_{\boldsymbol{x}}\{\boldsymbol{y}^T\boldsymbol{x} - \phi(\boldsymbol{x})\}$ and the subdifferential of ϕ at $\bar{\boldsymbol{x}} \in \operatorname{dom}\phi$ is $\partial\phi(\bar{\boldsymbol{x}}) := \{\boldsymbol{v} \,|\, g(\boldsymbol{x}) \geq g(\bar{\boldsymbol{x}}) + \boldsymbol{v}^T(\boldsymbol{x} - \bar{\boldsymbol{x}})\ (\boldsymbol{x}\in\operatorname{dom}\phi)\}$.

Given two lower semicontinuous convex functions with nonempty domain $k : \mathbb{R}^n \to \mathbb{R}\cup\{+\infty\}, h : \mathbb{R}^m \to \mathbb{R}\cup\{+\infty\}$, a matrix $A \in \mathbb{R}^{m\times n}$ and $\boldsymbol{b}\in\mathbb{R}^m$ this duality scheme makes a connection between the optimization problem

$$\min_{\boldsymbol{x}}\ k(\boldsymbol{x}) + h(\boldsymbol{b} - A\boldsymbol{x}), \tag{6}$$

called the *primal* problem, with its associated *dual* problem

$$\max_{\boldsymbol{y}}\ \boldsymbol{b}^T\boldsymbol{y} - k^*(A^T\boldsymbol{y}) - h^*(\boldsymbol{y}). \tag{7}$$

Fenchel-Rockafellar duality now states that, under the qualification condition

$$\boldsymbol{b} \in \operatorname{int}\,(A\mathrm{dom\ k} + \mathrm{dom\ h}),$$

the optimal value of the primal and dual problem coincide and that, given a solution $\bar{\boldsymbol{y}}$ of the dual problem, a solution of the primal can be recovered from the relation $\bar{\boldsymbol{x}} \in \partial k^*(A^T\bar{\boldsymbol{y}})$. We will from now on refer to (4) as the primal problem. To apply the Fenchel-Rockafellar scheme, we need to compute the conjugates of the functions in play. The conjugate of $\mathscr{K}(\cdot;\mu)$ can be computed by considering the $\log\exp$ function $\log\exp : \boldsymbol{y} \mapsto \log\left(\sum_{i=1}^n \exp(y_i)\right)$ and noting that

$$\log\exp^*(\boldsymbol{q}) = \begin{cases} \sum_{i=1}^n q_i\log(q_i) & \text{for } \boldsymbol{q}\in\Delta_n \\ +\infty & \text{otherwise} \end{cases},$$

as discussed in [45, Ex. 11.12]. Observe that we can express the Kullback-Leibler entropy as

$$\mathscr{K}(\boldsymbol{p};\mu) = \sum_{i=1}^{2^N} p_i\log(p_i) - \langle\boldsymbol{p}, \log\mu\rangle.$$

As $\log\exp$ is finite-valued and convex (hence lower semicontinuous and proper), the Fenchel-Moreau theorem (see e.g., [45, Theorem 11.1]) yields $\log\exp = (\log\exp^*)^*$.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

Therefore, also using [45, Eq. 11(3)], the conjugate of $\mathscr{K}(\cdot;\mu)$ is given by

$$\mathscr{K}^*(\boldsymbol{q};\mu) = \log\left(\sum_{i=1}^{2^N} \mu_i \exp\,(q_i)\right). \tag{8}$$

The same reference [45, Eq. 11(3)] together with [45, Ex. 11.11] also gives

$$\left(\frac{\alpha}{2}\|\cdot\|^2\right)^* = \frac{1}{2\alpha}\|\cdot\|^2 \quad (\alpha > 0).$$

We now obtain the dual problem by setting $k = \mathscr{K}(\,\cdot\,,\mu),\ h = \frac{\alpha}{2}\|\cdot\|^2$. Hence

$$\sup_{\boldsymbol{\lambda}}\left\{\langle \boldsymbol{b},\boldsymbol{\lambda}\rangle - \left(\tfrac{\alpha}{2}\|\cdot\|^2\right)^*(\boldsymbol{\lambda}) - \mathscr{K}^*(M^T\boldsymbol{\lambda};\mu)\right\},$$

is the resulting dual problem with $\boldsymbol{\lambda}$ denoting the dual variable, $\boldsymbol{b}$ the acquired image, $M = CA$ and μ the prior. Substituting the conjugates computed previously into this expression, this problem can be written explicitly as

$$\sup_{\boldsymbol{\lambda}}\left\{\langle \boldsymbol{b},\boldsymbol{\lambda}\rangle - \tfrac{1}{2\alpha}\|\boldsymbol{\lambda}\|^2 - \log\left(\sum_{i=1}^{2^N}\mu_i \exp\left(M_i^T\boldsymbol{\lambda}\right)\right)\right\}. \tag{9}$$

We note that, on $\mathbb{R}^m$, the domain of $\frac{\alpha}{2}\|\cdot\|^2$ is the entire space, so

$$\boldsymbol{b} \in \operatorname{int}\left(M\,\operatorname{dom}(\mathscr{K}) + \operatorname{dom}\left(\frac{\alpha}{2}\|\cdot\|^2\right)\right),$$

is trivially satisfied. This condition ensures that the optimal value of (4) is attained for at least one $\bar{\boldsymbol{p}} \in \Delta_{2^N}$, this is a property of the duality scheme that has been used. Note moreover that $\bar{\boldsymbol{p}}$ is guaranteed to be an element of the unit simplex as otherwise the Kullback-Leibler divergence takes on a value of infinity. Similarly, since

$$\boldsymbol{0} \in \operatorname{int}\left(M^T\,\operatorname{dom}\left(\frac{\alpha}{2}\|\cdot\|^2\right)^* - \operatorname{dom}(\mathscr{K}^*)\right),$$

the optimal value of (9) is also attained for at least one $\bar{\boldsymbol{\lambda}}$. Together, these conditions ensure that these problems share the same finite optimal value. Moreover, given a solution $\bar{\boldsymbol{\lambda}}$ of (9) one can perform primal-dual recovery via

$$\bar{\boldsymbol{p}} = \nabla\mathscr{K}^*(M^T\bar{\boldsymbol{\lambda}};\mu), \tag{10}$$

which is another property of this duality scheme. The previous equation is formulated in terms of the gradient, as $\mathscr{K}^*$ is differentiable at every point of its domain such that its subgradient at a given point is a singleton, namely its gradient at that point by [46, Theorem 25.1].

One of the advantages of this dual formulation is that solving the primal problem, a minimization problem in $\boldsymbol{p} \in \mathbb{R}^{2^N}$, is now analogous to solving the dual problem, a maximization problem in $\boldsymbol{\lambda} \in \mathbb{R}^{Nm}$ and recovering a solution to the primal problem via (10). This foray into Fenchel-Rockafellar duality has therefore yielded a tremendous dimensionality reduction. Despite this amelioration, solving the dual problem is still intractable as the conjugate of the entropy contains an immense sum over 2^N elements and the matrix M has dimensions $Nm \times 2^N$. This matrix cannot feasibly be stored in memory for large N.

2.1.2 Exploiting the Probabilistic Structure of the Dual Problem

Recall that, by definition, $M = CA$ where A is a $N \times 2^N$ matrix whose columns consist of the binary sequences of length of N. In particular, A_i^T is the ith binary sequence in some arbitrarily chosen ordering of $\{0,1\}^N$. The sum in (9) can therefore be rewritten as

$$\sum_{i=1}^{2^N}\mu(A_i^T)\exp(\langle A_i^T, C^T\boldsymbol{\lambda}\rangle). \tag{11}$$

This expression equals $\mathbb{E}_\mu\left[\exp\langle C^T\boldsymbol{\lambda},\boldsymbol{X}\rangle\right]$ with $\mathbb{E}_\mu[h(\boldsymbol{X})]$ denoting the expected value of the random variable $h(\boldsymbol{X})$, where $\boldsymbol{X}$ has probability distribution μ [47, Definition 1 p.141]. This expectation is simply the moment generating function (MGF) $M_{\boldsymbol{X}}$ of $\boldsymbol{X}$ evaluated at $C^T\boldsymbol{\lambda}$. By assumption, $\boldsymbol{X} = (X_1,\ldots,X_N)$ where the X_i are independent Bernoulli random variables. Therefore, using [47, Theorem 5 p.155], the MGF in (11) can be written as

$$\prod_{i=1}^{N} M_{X_i}(C_i^T\boldsymbol{\lambda}). \tag{12}$$

The MGF of a Bernoulli random variable is made explicit in [47, Section 5.2.2 p.180]. Hence (12) is equivalent to

$$\prod_{i=1}^{N}(1-\rho_i+\rho_i\exp(C_i^T\boldsymbol{\lambda})),$$

where ρ_i is the probability that the ith bar in $\boldsymbol{x}$ is white. Replacing the sum in (8) with this product yields the following expression for the conjugate of the Kullback-Leibler divergence:

$$\mathscr{K}^*(M^T\boldsymbol{\lambda};\mu) = \sum_{i=1}^{N}\log\left(1-\rho_i+\rho_i\exp(C_i^T\boldsymbol{\lambda})\right). \tag{13}$$

This expression is easily evaluated given some $\boldsymbol{\lambda}$. Using this form for $\mathscr{K}^*$ renders the dual problem (9) tractable via standard numerical optimization algorithms. However, we recall that $\bar{\boldsymbol{p}} \in \Delta_{2^N}$, hence determining an expression for $\bar{\boldsymbol{p}}$ is infeasible regardless of the fact that we can solve the dual problem. We opt therefore to recover the original image directly from $\bar{\boldsymbol{\lambda}}$.

2.1.3 Determining the Original Image from the Argmax of the Dual Problem

In the following we seek to compute the expectation of (10) which serves as the estimate of the original image. Performing this calculation naively leads to (14) which includes the large matrix M. Thus we use the probabilistic argument of the previous section to derive an analogous expression (15) which can be computed explicitly.

Given an optimal solution $\bar{\boldsymbol{p}}$ of the primal problem (4), we can recover an estimate of the original image $\boldsymbol{x}$ via $\bar{\boldsymbol{x}} = A\bar{\boldsymbol{p}}$. We refer to $\bar{\boldsymbol{x}}$ as an estimate of $\boldsymbol{x}$, as the penalty

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

function (2) does not guarantee that $C\bar{\boldsymbol{x}} = \boldsymbol{b}$. Using our expression from (10), we can write

$$\bar{x} = A\nabla\mathscr{K}^*(M^T\bar{\boldsymbol{\lambda}};\mu).$$

We first write $\nabla\mathscr{K}^*(M^T\bar{\boldsymbol{\lambda}};\mu)$ componentwise yielding

$$\left[\nabla\mathscr{K}^*(M^T\bar{\boldsymbol{\lambda}};\mu)\right]_k = \begin{cases} \frac{\mu_k \exp(M_k^T\bar{\boldsymbol{\lambda}})}{\sum_{i=1}^{2^N}\mu_i \exp(M_i^T\bar{\boldsymbol{\lambda}})} & \text{for } k\in I,\\ 0 & \text{otherwise,}\end{cases}$$

such that $\bar{x}$ can be written componentwise by multiplying the previous expression by A, i.e.

$$\bar{x}_k = \frac{\sum_{i=1}^{2^N} a_{ki}\mu_i \exp(M_i^T\bar{\boldsymbol{\lambda}})}{\sum_{i=1}^{2^N}\mu_i \exp(M_i^T\bar{\boldsymbol{\lambda}})}. \tag{14}$$

Here a_{ij} is the value in the ith row of the jth column of A. We now consider

$$\nabla \log\left(\sum_{i=1}^{2^N}\mu_i \exp\langle A_i^T,\cdot\rangle\right),$$

the kth component of which is simply

$$\frac{\sum_{i=1}^{2^N} a_{ki}\mu_i \exp\langle A_i^T,\cdot\rangle}{\sum_{i=1}^{2^N}\mu_i \exp\langle A_i^T,\cdot\rangle}.$$

Evaluating this expression at the point $C^T\bar{\boldsymbol{\lambda}}$ demonstrates that it is equivalent to (14) (since $M = CA$) with the advantage that we can simplify it using the same probabilistic argument that was derived previously. Thus, x can be estimated whilst bypassing the matrix A via

$$\bar{\boldsymbol{x}} = \nabla\sum_{i=1}^{N}\log\left(1-\rho_i+\rho_i \exp(C_i^T\bar{\boldsymbol{\lambda}})\right). \tag{15}$$

Consequently, once the argmax of the dual problem has been determined we can estimate the original image directly, without determining A or the PMF $\bar{\boldsymbol{p}}$.

Each step of the image estimation has now been made computationally tractable.

2.1.4 A Summary of the Steps for Image Estimation

The previous developments can be summarized in the following procedure for deblurring an image for which the convolution kernel is known or approximated.

First, a prior μ must be formed. This prior will assign a probability of being white to each bar, so encoding the symbolic constraints of the barcode of interest into the prior will ensure that the solution to (4) is at least correct on these bars. Other types of priors and a more detailed discussion of the construction of this symbolic prior is found in the results section.

Next, the dual problem (9) with the expression for the conjugate of the Kullback-Leibler divergence given in (13) is solved. This step can be performed efficiently by standard optimization software. Our choice of algorithm is discussed in the results section.

Finally, an estimate of the initial image is determined via (15). The resulting image will not be identical to x due to rounding errors and the choice of tolerance in the optimization algorithm.

We choose to subsequently perform a thresholding step to guarantee that all of the segments of the barcode are either 0 or 1. This step ensures that the barcode will be readable if it was accurately deblurred and thus the information encoded in the original image can be extracted if our method has succeeded.

2.2 The Kernel Estimation

We now focus on solving (5), keeping in mind that it shares a similar paradigm to (4). Again, since the convolution is linear and discrete, $\boldsymbol{c} * UA\boldsymbol{p}$ can be written as $X\boldsymbol{c}$. We enforce that $\boldsymbol{c}\in\mathbb{R}^{Nm}$, as the convolution kernel should not be larger than the size of the image. Thus, $X\in\mathbb{R}^{95m\times 95m}$ such that (5) can be solved directly as a constrained minimization problem, since it is not as high-dimensional a problem as (4). However mimicking the previous foray into Fenchel-Rockafellar duality will yield a simpler unconstrained analogue to this primal problem.

2.2.1 Advantages of the Dual Formulation

The dual problem to (5) is nearly identical to (9), hence we simply state the dual problem using the same duality scheme

$$\sup_{\boldsymbol{\lambda}}\left\{\langle\boldsymbol{b},\boldsymbol{\lambda}\rangle - \tfrac{1}{2\beta}\|\boldsymbol{\lambda}\|^2 - \mathscr{K}^*(X^T\boldsymbol{\lambda};\nu)\right\}. \tag{16}$$

The same argument used to show that (4) and (9) share the same optimal value and that this solution is attained in both problems implies that (5) and (16) satisfy the same property. Consequently, the argmin $\bar{c}$ of (5) is given by

$$\bar{\boldsymbol{c}} = \nabla\left(\log\left(\sum_{i=1}^{95m}\nu_i \exp(X_i^T(\cdot))\right)\right)(\bar{\boldsymbol{\lambda}}), \tag{17}$$

in the same vein as (10). Here, $\bar{\boldsymbol{\lambda}}$ denotes the argmax of the unconstrained dual problem.

Regularizing problem (5) via a Kullback-Leibler divergence term guarantees that the optimal kernel estimate $\bar{c}$ is nonnegative and that its elements sum to 1 as explained. This property is characteristic of any normalized blur kernel which is precisely the type of PSF that occurs in image acquisition. Moreover, ν is used to limit the size of the considered kernel by setting all but a square of the desired width centred at the middle of its matrix representation to 0 and setting uniform values summing to 1 in this square. Hence, adopting a coarse-to-fine approach as in [18], [48] is analogous to simply increasing the size of the kernel being considered at each step which can be accomplished by varying ν.

2.3 The Algorithm

We summarize the development of the prior probability distributions and outline an algorithm that implements our blind deblurring method.

Barcode symbologies impose constraints which typically fix certain segments of an image. We outline a method to generate a prior which captures these constraints. Recall that we have modeled a barcode by a vector of N independent Bernoulli random variables. The distribution of a Bernoulli random variable is completely determined by a single parameter ρ. As a barcode is a vector of independent

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

Bernoulli random variables, its distribution is determined by N parameters ρ_i as in Eq. (13) above, where each ρ_i represents the probability that the ith bar in a barcode is white. This suggests that a natural prior distribution μ has the following probability mass function

$$\mu(\boldsymbol{x}) = \prod_{i=1}^{N} \rho_i^{x_i}(1-\rho_i)^{1-x_i},$$

where x_i is the ith bar of $\boldsymbol{x}$. We let $\rho_i = 0$ if the ith bar is fixed as black by the barcode symbology, $\rho_i = 1$ if the ith bar is fixed as white, and $\rho_i = \frac{1}{2}$ if the ith bar is not fixed by the symbology. (This choice reflects our lack of prior knowledge of the state of the ith bar when it is not fixed by a symbology.)

The algorithm is summarized with references to the relevant equations in Algorithm 1. The algorithm features two loops. The outer loop iterates through a set of fixed widths for our kernel estimate. The inner loop repeatedly solves problems (4) and (5).

We begin by setting $i = 1$ and hence initially assume that the size of the convolution kernel is $2i + 1 = 3$. We take our initial best guess of the true barcode to be the image which is black or white in regions which are fixed as such by the relevant barcode symbology, and gray in all other regions. (See the lower image in Fig. 1 for an example of such an initial guess for UPC barcodes.) We substitute this image for $A\bar{\boldsymbol{p}}$ in the kernel estimation step (5), solve its dual problem (16), and then compute (17) to obtain our initial estimate $\bar{\boldsymbol{c}}$ of the true convolution kernel. We then use this estimate $\bar{\boldsymbol{c}}$ to solve the image estimation problem (4) via the methods outlined in the previous sections and obtain a first estimate $\bar{\boldsymbol{x}}$ of the true barcode.

This estimated barcode is subsequently read by a software barcode scanner. If the barcode is readable, the algorithm terminates successfully. If the algorithm does not terminate after the first iteration of the inner loop, we continue to iterate through the alternating kernel estimation and image estimation steps, each time substituting our image estimate for $A\bar{\boldsymbol{p}}$ in the kernel estimation step, and our subsequent kernel estimate for $\boldsymbol{c}$ in the image estimation step. This yields progressive improvements in the estimates of the convolution kernel and the image.

Algorithm 1. Entropic Blind Deblurring

Input: Blurred image $\boldsymbol{b}$, prior μ
  for $i = 1$ to (width of b)$/2$, **do**
    $\bar{\boldsymbol{x}} \leftarrow \mu$
    width $\leftarrow (2i+1)$
    for $j = 1$ to 5 **do**
      $\bar{\boldsymbol{c}} \leftarrow (17)$ at $\arg\max$ of (16) with size width
      $C \leftarrow \bar{\boldsymbol{c}}$ written as a convolution matrix
      $\bar{\boldsymbol{x}} \leftarrow (15)$ at $\arg\max$ of (9) using expression (13)
      threshold $\bar{\boldsymbol{x}}$
      if $\bar{\boldsymbol{x}}$ is readable **then**
        return $\bar{\boldsymbol{x}}$
      end if
    end for
  end for
  return $\bar{\boldsymbol{x}}$
Output: approximated image $\bar{\boldsymbol{x}}$

Fig. 2. A graphical depiction of the types of kernel used with width 5. The left kernel generates Gaussian blur and the right one generates linear motion blur at an angle of $\frac{\pi}{4}$. Note that they are normalized such that the intensity values sum to 1.

If the barcode is still not readable after a fixed number of iterations (5 in our implementation), we infer that the initial width $2i + 1$ of the estimated convolution kernel was too small. We therefore increment i and iterate through the inner loop once again. We iterate through the outer loop until the width of the convolution kernel reaches that of the image. If the barcode is still not readable at this point, the algorithm terminates unsuccessfully.

We have therefore set up a framework that permits blind deblurring for both QR and UPC-A barcodes that effectively utilizes prior knowledge of their respective structure.

3 Results

In the following, we will discuss some of the results obtained while testing our method. We will refer to graphs in the online supplementary material, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPAMI.2019.2927311, for more details of our experiments. First, we explain the methodology used to generate all of the relevant quantities used for testing.

3.1 Implementation Details

We began by generating barcodes in both the UPC-A and QR symbologies. In the case of UPC-A barcodes, we considered 200000 valid barcodes from the Open Product Data database [36]. In the case of QR codes, we chose various phrases to encode and used an online QR code generator to get the relevant images. Each phrase was encoded in all four levels of error tolerance supported by the symbology, namely low (7 percent tolerance), medium (15 percent tolerance), quartile (25 percent tolerance) and high (30 percent tolerance) as explained in [42].

In order to blur images synthetically, normalized PSFs were generated. For Gaussian convolution kernels, this process is straightforward in both the 1D and 2D cases, as for a PSF of width k one need only sample a 1D or 2D Gaussian function with mean 0 at k points on an interval centred at 0. For box blurs in the 1D case, we simply initialize all k points of the PSF to the value $\frac{1}{k}$. For linear motion blurs in the 2D case, the motion blur is simply a line through the centre of the kernel at a prescribed angle. Examples of these kernels are compiled in Fig. 2.

Moreover, general motion blur kernels such as those in [30] were tested on QR codes, yielding adequate results on barcodes of a reasonable size relative to the kernel as demonstrated in Fig. 3 (other examples are presented in Fig. Sup. 1 of the supplementary material, available online). In what follows we concentrate on Gaussian and linear motion blur as their testing can be readily automated.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

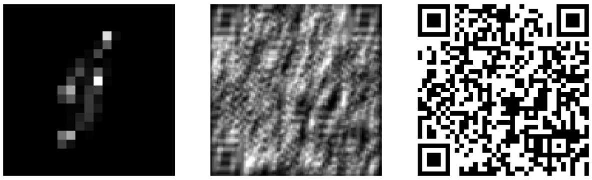

Fig. 3. The leftmost image is a general motion blur kernel. The center image is the corresponding blurred QR code. The rightmost image, which is readable, is the result obtained by applying our method. The kernel was normalized such that its intensity values summed to 1 prior to blurring and the QR code was upscaled by a factor of 3. The size of the barcode image is 111×111 pixels and the size of the kernel is 19×19.

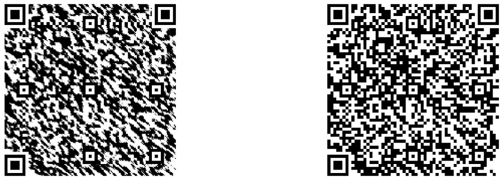

Fig. 5. This figure demonstrates the utility of the thresholding step. The left hand side is the deblurred image prior to the downscaling and rounding. The original image was subjected to linear motion blur with large kernel size at an angle of $-\frac{\pi}{4}$. Note that some degree of distortion along a diagonal axis remains prior to thresholding and downscaling. The right side displays the barcode post-thresholding; it is readable.

The barcodes are upscaled prior to convolution in order to access a greater range of blurring magnitudes. Indeed, even blur kernels of width 3 (the smallest tested) produce dramatic quantities of blur when one bar is the size of one pixel as demonstrated in Fig. 4. Thus, upscaling allows us to consider more realistic levels of blurring.

In preliminary testing, the PSF was turned into a convolution matrix C by examining the result of the discrete convolution as demonstrated in [17, Fig. 4.7] and inferring a matrix that performs the same operation. A similar method was used to determine the matrix X in (16). We worked with the 0 boundary conditions in the formation of these matrices, as it is simpler to construct the convolution matrix in this case. In reality the barcode is encased in a white quiet zone so one can simply invert the colours of the captured signal such that the 0 boundary condition in the inverted image is equivalent to a 1 boundary condition in the original. Hence, in a real image, the 0 boundary condition would be sufficient, as the barcode would not be convolved with data outside this quiet zone for blur kernels of a reasonable size.

Thereafter, the convolutions were performed by means of the fftconvolve method from the scipy python library [15]. The advantage to this approach is that it is both faster and less memory intensive than forming the convolution matrix and storing it in memory. We make precise that the Hermitian adjoint of the discrete convolution operator is obtained by performing a discrete convolution with one of the arrays reversed about its axes as discussed in [9, Section 5.1.1]. Hence, passing from the matrix methodology to this one is akin to replacing the transposed matrices in the dual problems and the recovery of the solutions to the primal problems by the adjoint of the corresponding convolution.

We employ a downscaling step once we have estimated our image by averaging together the blocks of pixels that correspond to one pixel once upscaled. We subsequently round the pixel intensities to the nearest integer as discussed previously. The utility of this step is highlighted in Fig. 5. The critical task of decoding the QR estimate is delegated to the Zbar Python implementation provided by [57] which permits automation for checking readability of the iterates during testing. Hence, if a barcode is readable post thresholding, we terminate the algorithm and return the data which has been decoded. We use both Zbar and various smartphone applications in order to compare the performance of our algorithm to state of the art QR code scanners.

We equally examine how our method performs in the presence of noise. We consider both additive Gaussian and salt and pepper noise throughout, as they are the most common in practice (cf. Fig. 6). To generate Gaussian noise, we generate a matrix of the same size as the image to which each pixel is associated a random sample from a normal distribution with prescribed variance and simply add both matrices to add noise to the image. Similarly, to generate salt and pepper noise of a given percentage, we generate at each pixel a random real number between 0 and 1. If this number is higher than our prescribed percentage, we add nothing to the image. If it is lower, we randomly choose between 0 or 1; if 0 is selected, the pixel is made black in the image, if 1 is selected, it is made white. Visuals are provided below to better illustrate the magnitude and types of noise.

All that remains before our algorithm can be tested is to make explicit how the various priors are generated in the different symbologies. Without considering the intricacies of the various encoding schemes, it is possible to form a uniform prior in which every bar is given a probability of 0.5 of being white. It is obvious that such a prior will not perform as well as a symbolic one which encodes all of the fixed modules within a symbology and assigns a uniform probability of 0.5 to the bars that have not been fixed. Note that QR codes have varying sizes, thus a prior must be generated for the various sizes. In the UPC-A case, a third prior was equally constructed in which our library of more than 200000 UPC-A barcodes was analyzed and each bar was given a probability reflecting the percentage of barcodes of the library having

Fig. 4. The image on the left presents a QR code which has not been upscaled prior to being blurred by a 3x3 Gaussian blur kernel. Note that this magnitude of blur is rather large, hence the need for upscaling the image prior to convolution. The right hand side is the result we obtain upon applying our blind deblurring algorithm with a symbolic prior. The right hand QR code can be read by any conventional QR code reader.

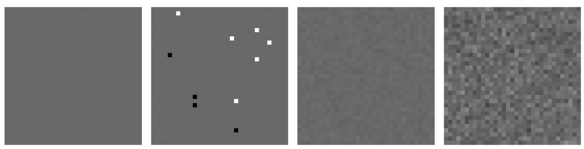

Fig. 6. A demonstration of types and magnitudes of noise tested. The leftmost image represents the noiseless case, the centre-left image depicts 1 percent salt and pepper noise, the centre-right image is 0.01 variance Gaussian noise and the rightmost image is 0.05 variance Gaussian noise.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

TABLE 1
This Table Compares the Performance of the Various Priors in the Presence of Both Types of Noise

Prior	Type of Blur	Cut-off Width
Uniform	Gaussian	173.0
	Box	129.8
Symbolic	Gaussian	259.8
	Box	210.6
Empirical	Gaussian	259.4
	Box	259.4

The cut off width is the width of the kernel at which the method first fails.

that bar white, this prior is referred to as empirical. Testing for UPC-A barcodes was performed first and it was deemed that the symbolic and empirical priors yield similar performance. As no tangible performance improvements were expected, we did not construct an empirical prior for QR codes.

With this framework in place to generate blurred and noisy barcodes, we are ready to test the performance of our method.

3.2 Non-Blind Deblurring

As mentioned previously, non-blind deblurring can be done by performing the image estimate step with the exact kernel c known. We wish to determine the performance of our method for this step and examine the effects of prior choice. Moreover, we wish to quantify the flexibility of our method with respect to the presence of noise in the acquired image.

3.2.1 Non-Blind Deblurring for UPC-A Barcodes

In order to gauge the performance of this method for non-blind deblurring of 1D barcodes, we begin by observing its noiseless performance.

We do so by choosing 5 random barcodes, upscaling them by a factor of 5 and blurring them with progressively larger blur kernels until the barcode was no longer successfully readable when using the method. We repeat this process for every prior as well as with both Gaussian and box blurs. The results of the five barcodes are then averaged in order to provide a general idea of the non-blind, noiseless performance. We set $\alpha = 1000000$ in order to give great importance to the error term thus incentivizing the proximity of $C\mathbb{E}_{\boldsymbol{p}}[\boldsymbol{x}]$ to $\boldsymbol{b}$. The results of this test are shown in Table 1.

We note that the empirical prior outperforms the symbolic prior in the case of box blur specifically and that they both outperform the uniform prior by a significant margin. Clearly, the structure encoded in the non-uniform priors account for their superior performance. Moreover, despite the assumption that the empirical prior should encode some form of correlation between the various bars, it performs essentially identically to the symbolic prior. Therefore, the intrinsic symbology of the barcode appears to take precedence over any additional structure gained by a statistical learning approach. Finally, the blur widths at which these priors first fail are so large that they would not occur in real life applications, this is a testament to the strength of our method in the case of noiseless image acquisition for UPC-A barcodes.

TABLE 2
This Table Compares the Performance of the Various Error Tolerances in QR Codes in the Presence of Different Types of Blur

Error Tolerance	Type of Blur	Cut-off Width (ZBar)	Cut-off Width (Ours)
Low	Gaussian	5.0	29.4
	Motion	6.6	30.6
Medium	Gaussian	5.4	32.2
	Motion	7.0	37.8
Quartile	Gaussian	5.4	33.8
	Motion	7.0	50.2
High	Gaussian	5.8	35.4
	Motion	7.0	65.8

The cut off width is the width of the kernel for which the method first fails.

As for the performance of this method as it pertains to a noisy image acquisition process, we determine a cutoff variance for Gaussian noise for various blur widths before which we can read all of the blurred and noisy barcodes generated. Again, we pick 5 random barcodes and begin by blurring with blur width 3 Gaussian noise. Next, we iteratively increase the variance of the additive Gaussian noise that is added to the image until we first fail to successfully deblur the barcode. We then increase the blur width by 2 and repeat this procedure until we reach a width such that even the lowest variance noise (0.005) cannot be read. At this point, we repeat the entire process with a box blur.

We note that salt and pepper noise was not tested for the UPC-A symbology, as in our one-dimensional formulation, this type of noise is equivalent to changing the color of the entire bar. In practical applications this noise would only effect a segment of a bar which our model is not designed to account for.

In these tests, $\alpha = 1000$ in order to account for the fact that $\boldsymbol{b}$ will not be in the range of C. These tests are performed with the three different priors and the results are compiled in Fig. Sup. 2 of the supplementary material, available online. We note again that the symbolic and empirical priors outperform the uniform prior by a significant margin for larger blur widths.

We note again that the symbolic and empirical priors outperform the uniform prior by a significant margin for larger blur widths.

3.2.2 Non-Blind Deblurring for QR Codes

We test our method for QR codes by picking five of the encoded messages and determining the blur width at which our method first fails to recover the information contained in the QR code. The barcodes are upscaled by a factor of 3 in order to consider a greater range of blurring kernels. We proceed similarly to the UPC-A testing, however, rather than considering different types of priors, we compare the different levels of error tolerance and use only a symbolic prior. We equally compare our method to the ZBar algorithm by attempting to read the blurred barcode prior to deblurring it. If it fails to read, we deblur the barcode using our method and verify if the image is now readable. Throughout these

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

TABLE 3
This Table Compiles the Blur Widths at Which Our Method First Fails to Recover the Information Contained in the UPCA Barcode When Using the Blind Deblurring Method

Prior	Type of Blur	Cut off Width
Uniform	Gaussian	19.4
	Box	19.0
Symbolic	Gaussian	76.6
	Box	75.0
Empirical	Gaussian	74.2
	Box	89.0

tests, we set $\alpha = 10000000$ in order to enforce the constraint on the mean. The averaged results of this testing are shown in Table 2.

We note in particular that the algorithm performs noticeably better in the presence of motion blur as compared to Gaussian blur. Letting l denote the width of the blur, motion blur kernels yield convolutions such that the value at one point is determined by the values of l points, whereas in the Gaussian case, l^2 points are considered. Hence, Gaussian blurs produce more dramatic blurring for the same size of kernel, so this observation is reasonable. The various error tolerances perform as expected, with the low tolerance performing the worst and the high tolerance performing the best. We note, moreover, that with every error tolerance, we are able to successfully recover images that are blurred far beyond what can be considered normal for real life applications.

In order to gauge the effects of additive noise on this method, we again approach the problem in a similar fashion to the 1-dimensional problem. The effects of salt and pepper noise can now be studied, as QR codes include a degree of error correction. Hence, even if the noise does not permit us to reconstruct the same QR code as the original, the deblurred image may still be read. We compile our results in Fig. Sup. 3 of the supplementary material, available online, here α was set at 750 to promote flexibility.

We note that, as expected, our method is more robust to Gaussian noise as it is less dramatic, changing the value of almost every module by a small amount as opposed to salt and pepper noise which makes a certain number of modules black or white. In comparing the various error tolerances, it is clear that the low level performs much worse than the others which perform almost identically. Regardless, the amounts of noise considered are far larger than those that would occur in realistic applications hence, this method performs adequately for reasonable noise levels.

Finally, the method employed by ZBar performs exceptionally well for very small quantities of blur. However our method greatly outperforms it for every blur width other than the smallest one.

TABLE 4
This Table Compares the Performance of the Various Error Tolerances in QR Codes in the Presence of Different Types of Blur

Error Tolerance	Type of Blur	Cut-off Width (ZBar)	Cut-off Width (Ours)
Low	Gaussian	5.0	9.0
	Motion	6.6	25.0
Medium	Gaussian	5.0	10.2
	Motion	7.0	27.0
Quartile	Gaussian	5.4	10.2
	Motion	7.0	33.8
High	Gaussian	5.8	10.6
	Motion	7.0	31.8

The cut off width is the width of the kernel for which this blind method first fails.

3.3 Blind Deblurring

We now test the performance of our method as it pertains to the problem of blind deblurring.

3.3.1 Blind Deblurring for UPC-A Barcodes

We wish to quantify the performance of our blind deblurring method in the presence of additive noise at various blur widths. We employ an identical series of tests to those used in the non-blind case. The only difference is that we utilize our non-blind method to deblur the resulting images. The noiseless results are shown in Table 3.

Comparing these results to those obtained with the non-blind method reveals, unsurprisingly, that the blind method is not as robust to the size of the blur width. This is to be expected, as the blur kernel is being estimated rather than provided. However, we note that these results are certainly adequate for real life conditions.

We equally examine the case in which noise is present in the image acquisition process by performing tests identical to those used in the non-blind section. The results are compiled in Fig. Sup. 4 of the supplementary material, available online.

We note that the symbolic and empirical priors perform nearly identically. Notably, they greatly outperform their uniform counterpart. Moreover, this blind method actually appears to outperform the non-blind method. This slight improvement may be due to the fact that the non-blind method enforces the use of the original convolution kernel of which b is often not in the range of. The blind deblurring method offers greater flexibility in terms of the convolution kernel and hence with the thresholding step we employ, it is conceivable that performance is improved.

3.3.2 Blind Deblurring for QR Barcodes

In the noiseless case, we employ the same tests as we did in non-blind deblurring. The results are shown in Table 4.

As expected, the results of this method are not as good as those obtained with its non-blind analogue. They are however acceptable for real world image acquisition.

Fig. 7. This figure demonstrates the strength of our method even in the case where very large Gaussian blur is present. The right hand side is the result we obtain upon applying our blind deblurring algorithm with a symbolic prior. The right hand QR code can be read by any conventional QR code reader.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

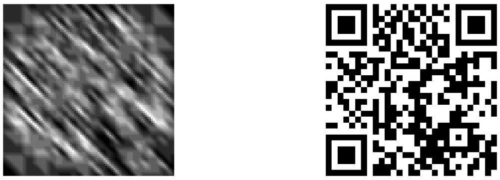

Fig. 8. This figure presents a blurred image on the left hand side. The blurring is a linear motion blur of kernel size 9 with angle $-\frac{\pi}{4}$ which has been performed on a 29×29 pixel QR code upscaled to 87×87 pixels. The right hand side is the result we obtain upon applying our blind deblurring algorithm with a symbolic prior. The right hand QR code can be read by any conventional QR code reader.

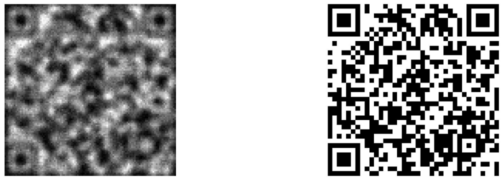

Fig. 10. On the left, a QR code with an upscaling factor of 3, subject to width 7 Gaussian blur with 0.05 variance Gaussian noise is presented. The right hand side is the result we obtain upon applying our blind deblurring algorithm with a symbolic prior. The right hand QR code can be read by any conventional QR code reader.

We equally perform the same tests as those considered in the non-blind case and compile the results in Fig. Sup. 5 of the supplementary material, available online.

The various levels of error tolerance perform as expected and we note that the flexibility with respect to both the Gaussian and salt and pepper noise is considerable, as they greatly surpass what could be considered reasonable at low widths of blur.

3.4 Possible Improvements

Given that our aim throughout was to explain our method and demonstrate the power of symbology in barcode reconstruction, we did not explore some avenues for improving it. We list some improvements for those who wish to implement them. First, we did not attempt to optimize solving either of the dual problems of interest, opting rather to implement the stock l-bfgs algorithm from the scipy package [56]. A further analysis of these two problems could yield a tailor-made approach to solving these problems that outperforms our current approach. This modification could significantly improve the run time of the algorithm, but would not likely improve its accuracy in terms of reproducing the original barcode.

Next, our preliminary tests were performed in the Python programming language using the Jupyter Notebook application as well as in the Matlab computing environment. All final testing was performed in Python, which is known to be slower than C/C++ especially as speed was not the main consideration when the code was written, as explained in [40]. One could conceivably decrease runtimes in a significant fashion by rewriting the code in a different language or simply by optimizing the code already written.

Moreover, the parameters α and β used during testing were determined empirically during testing. A more detailed analysis of these parameters may prove fruitful in enhancing the performance of this method depending on the context in which it is used.

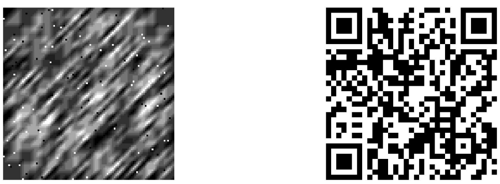

Fig. 9. On the left, a QR code with an upscaling factor of 3, subject to width 11 motion blur at an angle of $\frac{\pi}{4}$ with 1 percent salt and pepper noise is presented. The right hand side is the result we obtain upon applying our blind deblurring algorithm with a symbolic prior. The right hand QR code can be read by any conventional QR code reader.

Furthermore, recall the algorithm terminates either when the approximate barcode has been read or when the iterations terminate. Thus, no supplementary processing is performed on the intermediate approximations of the image before attempting to read them. In the case of UPC-A barcodes, we are simply verifying each segment against a dictionary of digits, hence if even a single bar fails the entire barcode will fail to read and the algorithm will continue to run. It would be possible to implement a method that would attempt to correct errors in the barcode prior to the reading step which could improve performance. Moreover, determining optimal values for the number of iterations to perform and verifying if one could increase the size of the blurring kernel quicker could significantly reduce runtime.

4 Conclusion and Real World Applications

Throughout this article, our focus has been on developing and testing a novel entropy-based method to solve a difficult ill-posed problem: blind and non-blind barcode deblurring of barcodes. The strength of our method is that it efficiently exploits the symbology innate to barcodes. Our results were tested on simulated images with moderate amounts of noise and large amounts of blurring. Moreover, many barcode reading software packages were considered. A natural question is to what extend our method can be used for real life camera images, i.e., industrial applications. Such applications are in no way immediate from our current set up. Note that our method depends heavily on the symbology and it is assumed that the scaling is uniform throughout the image; thus any implementation would require a significant amount of preprocessing to obtain data to which our algorithm can be directly applied. This dependence on symbology suggests that our method is ill-suited

Fig. 11. Out of focus image of a QR code.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

Fig. 12. Result of applying our method to a processed version of Fig. 11.

for situations where the blur is not uniform. While the general details of such preprocessing are beyond the scope of this article, we include, as proof of concept for the applicability of this method to real life situations, one example in Fig. 11. Here, we present a picture of a barcode with significant out of focus blur. Zbar and our smart phones are unable to read this picture. Fig. 12 is the readable barcode obtained by applying our method after isolating the barcode from the image. As explained in Section 3.1, the boundary conditions were accounted for by inverting the colours of the signal before the method was applied.

Acknowledgments

Gabriel Rioux and Christopher Scarvelis would like to acknowledge funding they received from the McGill SURA and NSERC USRA respectively which permitted their contribution to this research. Rustum Choksi and Tim Hoheisel were partially supported by NSERC (Canada) Discovery Grants. The authors would like to thank Jerome Gilles for helpful comments on a previous draft as well as the anonymous reviewers for their useful feedback. Gabriel Rioux and Christopher Scarvelis Fan are co-first authors. Rioux and Scarvelis equally share the role of first author.

References

[1] J. F. Bercher, G. Le Besnerais, and G. Demoment, *Maximum Entropy and Bayesian Methods*. Berlin, Germany: Springer, 1996, pp. 223–232.

[2] J. Borwein and A. S. Lewis, "Partially finite convex programming, part I: Quasi relative interiors and duality theory," *Math. Program.*, vol. 57, pp. 15–48, 1992.

[3] J. Borwein and A. S. Lewis, "Partially finite convex programming, part II: Explicit lattice models," *Math. Program.*, vol. 57, pp. 49–83, 1992.

[4] L. Chen, "A directed graphical model for linear barcode scanning from blurred images," in *Proc. 11th Asian Conf. Comput. Vis.*, 2013, pp. 524–535.

[5] R. Choksi, Y. van Gennip, and A. Oberman, "Anisotropic total variation regularized L^1-approximation and denoising/deblurring of 2D bar codes," *Inverse Problems Imag.*, vol. 5, no. 3, pp. 591–617, 2011.

[6] H. Cho, S. Lee, and J. Wang, "Text image deblurring using text-specific properties," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 524–537.

[7] S. Cho and S. Lee, "Fast motion deblurring," *ACM Trans. Graph.*, vol. 28, no. 5, pp. 145:1–145:8, 2009.

[8] C.-H. Chu, D.-N. Yang, and M.-S. Chen, "Image stabilization for 2D barcode in handheld devices," in *Proc. 15th Int. Conf. Multimedia*, 2007, pp. 697–706.

[9] J. F. Clærbout, *Earth Soundings Analysis: Processing versus Inversion*, 2nd ed. 2004. [Online]. Available: sep.stanford.edu/sep/prof/pvi.pdf

[10] M. Dodangeh, I. N. Figueiredo, and G. Gonçalves, "Spatially adaptive total variation deblurring with split Bregman technique," *IET Image Process.*, vol. 12, no. 6, pp. 948–958, Jun. 2018.

[11] D. Dacunha-Castelle and F. Gamboa, "Maximum d'entropie et problème des moments," *Annales de l'IHP Probabilités et Statistiques*, vol. 26, no. 4, pp. 567–596, 1990.

[12] O. Eisaku, H. Hiroshi, and A. H. Lim, "Barcode readers using the camera device in mobile phones," in *Proc. Int. Conf. Cyber Worlds*, 2004, pp. 260–265.

[13] S. Esedoglu, "Blind deconvolution of bar code signals," *Inverse Problems*, vol. 20, pp. 121–135, 2004.

[14] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman, "Removing camera shake from a single photograph," in *Proc. ACM SIGGRAPH 2006 Papers*, 2006, pp. 787–794.

[15] Fftconvolve. 2019. [Online]. Available: https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.fftconvolve.html

[16] Y. van Gennip, P. Athavale, J. Gilles, and R. Choksi, "A regularization approach to blind deblurring and denosing of QR barcodes," *IEEE Trans. Image Process.*, vol. 24, no. 9, pp. 2864–2873, Sep. 2015.

[17] P. Hansen, J. Nagy, and D. O'Leary, *Deblurring Images*, 3rd ed. Philadelphia, PA, USA: SIAM, 2006.

[18] M. Iwen, F. Santosa, and R. Ward, "A symbol-based algorithm for decoding bar codes," *SIAM J. Imag. Sci.*, vol. 6, no. 1, pp. 56–77, 2013.

[19] E. T. Jaynes, "Information theory and statistical mechanics," *Phys. Rev.*, vol. 106, no. 4, pp. 620–630, 1957.

[20] E. T. Jaynes, "Information theory and statistical mechanics. II," *Phys. Rev.*, vol. 108, no. 2, pp. 171–190, 1957.

[21] E. Joseph and T. Pavlidis, "Deblurring of bilevel waveforms," *IEEE Trans. Image Process.*, vol. 2, no. 2, pp. 223–235, Apr. 1993.

[22] E. Joseph and T. Pavlidis, "Bar code waveform recognition using peak locations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 630–640, Jun. 1994.

[23] N. Joshi, R. Szeliski, and D. J. Kriegman, "PSF estimation using sharp edge prediction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[24] H. Kato and K. T. Tan, "Pervasive 2D barcodes for camera phone applications," *IEEE Pervasive Comput.*, vol. 6, no. 4, pp. 76–85, Oct.–Dec. 2007.

[25] J. Kim and H. Lee, "Joint nonuniform illumination estimation and deblurring for bar code signals," *Opt. Exp.*, vol. 15, no. 22, pp. 14817–14837, 2007.

[26] J. Kotera, F. Šroubek, and P. Milanfar, "Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors," in *Proc. Int. Conf. Comput. Anal. Images Patterns*, 2013, pp. 59–66.

[27] S. Kresic-Juric, "Edge detection in bar code signals corrupted by integrated time-varying speckle," *Pattern Recognit.*, vol. 38, no. 12, pp. 2483–2493, 2005.

[28] S. Kresic-Juric, D. Madej, and F. Santosa, "Applications of hidden Markov models in bar code decoding," *Pattern Recognit. Lett.*, vol. 27, no. 14, pp. 1665–1672, 2006.

[29] S. Kullback and A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[30] A. Levin, Y. Weiss, F. Durand, and W. T. Freeman, "Understanding and evaluating blind deconvolution algorithms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1964–1971.

[31] S. Li, J. Shang, Z. Duan, and J. Huang, "Fast detection method of quick response code based on run-length coding," *IET Image Process.*, vol. 12, no. 4, pp. 546–551, 2018.

[32] N. Liu, X. Zheng, H. Sun, and X. Tan, "Two-dimensional bar code out-of-focus deblurring via the increment constrained least squares filter," *Pattern Recognit. Lett.*, vol. 34, pp. 124–130, 2013.

[33] N. Liu, Y. Du, and Y. Xu, "QR codes blind deconvolution algorithm based on binary characteristic and L0 norm minimization," *Pattern Recognit. Lett.*, vol. 111, pp. 117–123, 2018.

[34] P. Maréchal and A. Lannes, "Unification of some deterministic and probabilistic methods for the solution of linear inverse problems via the principle of maximum entropy on the mean," *Inverse Problems*, vol. 13, no. 1, pp. 135–151, 1997.

[35] P. Maréchal, "A note on entropy optimization," in *Approximation, Optimization and Mathematical Economics*. Berlin, Germany: Springer, 2001, pp. 205–211.

[36] Open Product Data Website. 2019. [Online]. Available: https://product-open-data.com/

[37] R. Palmer, *The Bar Code Book: A Comprehensive Guide to Reading, Printing, Specifying, Evaluating, and Using Bar Code and other Machine-Readable Symbols*, 5th ed. Bloomington, IN, USA: Trafford Publishing, 2007.

[38] J. Pan, Z. Hu, Z. Su, and M. Yang, "Deblurring text images via L0-regularized intensity and gradient prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2014, pp. 2901–2908.

[39] D. Parikh and G. Jancke, "Localization and segmentation of a 2D high capacity color barcode," in *Proc. IEEE Workshop Appl. Comput. Vis.*, 2008, pp. 1–6.

[40] Python Speed. 2017. [Online]. Available: https://wiki.python.org/moin/PythonSpeed

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.

[41] H. Pu, M. Fan, J. Yang, and J. Lian, "Quick response barcode deblurring via doubly convolutional neural network," *Multimedia Tools Appl.*, 2018. [Online]. Available: https://doi.org/10.1007/s11042-018-5802-2
[42] Wikipedia, "QR code." 2019. [Online]. Available: https://en.wikipedia.org/wiki/QR_code
[43] A. N. Rajagopalan and R. Chellappa, Eds., *Motion Deblurring: Algorithms and Systems*. Cambridge, U.K.: Cambridge Univ. Press, 2014.
[44] R. T. Rockafellar, *Conjugate Duality and Optimization*. Philadelphia, PA, USA: SIAM, 1974.
[45] R. T. Rockafellar and R. J.-B. Wets, *Variational Analysis*. Berlin, Germany: Springer-Verlag, 1998.
[46] R. T. Rockafellar, *Convex Analysis*, 2nd ed. Princeton, NJ, USA: Princeton Univ. Press, 1972.
[47] V. K. Rohatgi and A. K. Md. E. Saleh, *An Introduction to Probability and Statistics*, 2nd ed. Hoboken, NJ, USA: Wiley, 2001.
[48] L. Sun, S. Cho, J. Wang, and J. Hays, "Edge-based blur kernel estimation using patch priors," in *Proc. IEEE Int. Conf. Comput. Photography*, 2013, pp. 1–8.
[49] G. Sörös, S. Semmler, L. Humair, and O. Hilliges, "Fast blur removal for wearable QR code scanners," in *Proc. ACM Int. Symp. Wearable Comput.*, 2015, pp. 117–124.
[50] J. T. Thielemann, H. Schumann-Olsen, H. Schulerud, and T. Kirkhus, "Handheld PC with camera used for reading information dense barcodes," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit. Demonstration Program*, pp. 1–2, 2004.
[51] W. Turin and R. A. Boie, "Bar code recovery via the EM algorithm," *IEEE Trans. Signal Process.*, vol. 46, no. 2, pp. 354–363, Feb. 1998.
[52] W. Wang, C. Wu, and J. Deng, "A general selective averaging method for piecewise constant signal and image processing," *J. Sci. Comput.*, vol. 76, no. 2, pp. 1078–1104, 2018.
[53] H. Yang, C. Alex, and X. Jiang, "Barization of low-quality barcode images captured by mobile phones using local window of adaptive location and size," *IEEE Trans. Image Process.*, vol. 21, no. 1, pp. 418–425, Jan. 2012.
[54] W. Xu and S. McCloskey, "2D barcode localization and motion deblurring using a flutter shutter camera," in *Proc. IEEE Workshop Appl. Comput. Vis.*, 2011, pp. 159–165.
[55] J. Zhang, "An alternating minimization algorithm for binary image restoration," *IEEE Trans. Image Process.*, vol. 21, no. 2, pp. 883–888, Feb. 2012.
[56] C. Zhu, R. H. Byrd, and J. Nocedal, "L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization," *ACM Trans. Math. Softw.*, vol. 23, no. 2, pp. 550–560, 1997.
[57] ZBar bar code reader. 2011. [Online]. Available: http://zbar.sourceforge.net/

Gabriel Rioux received the undergraduate degree in mathematics and physics from McGill University, and he is working toward the master's degree in mathematics at McGill University, since the summer of 2019. His research area of interest is the interplay between optimal transport and partial differential equations.

Christopher Scarvelis is working toward the undergraduate degree in mathematics and economics at McGill University. His research interests include image processing and spectral graph theory.

Rustum Choksi received the PhD degree in mathematics from Brown University, in 1994. He held post-doctoral positions with the Center for Nonlinear Analysis, Carnegie Mellon University and the Courant Institute, New York University. From 1997 to 2010, he was a faculty member with the Department of Mathematics, Simon Fraser University. In 2010, he joined McGill University where he is currently a full professor with the Department of Mathematics and Statistics. His main research interests include interaction of the calculus of variations and partial differential equations with pattern formation.

Tim Hoheisel received the doctorate degree in mathematics from Julius-Maximilians University, Würzburg, in 2009. He held a postdoctoral position in Würzburg until 2016. During this time, he was a visiting professor with Heinrich-Heine University, Düsseldorf in 2011/12, and a visiting scholar with the University of Washington, Seattle, in 2012 and 2014. He joined the Department of Mathematics and Statistics, McGill University, Montreal, in 2016 as an assistant professor. His research area lies at the interface of continuous optimization and nonsmooth analysis.

Pierre Maréchal received the PhD degree in signal and image processing from the Universty of Toulouse. He was trained as an engineer. After a two-year postdoc with Simon Fraser University, he became assistant professor with the University of Montpellier. Since 2005, he is full professor with the University of Toulouse. His main research interests include inverse problems, optimizations, and applications.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/csdl.

Authorized licensed use limited to: McGill University. Downloaded on December 07,2020 at 14:09:40 UTC from IEEE Xplore. Restrictions apply.