Uniqueness of Solutions to Circular Deconvolution

Christopher Liu

August 16, 2013

1 Abstract

My research goes into the theoretical side of barcode recovery, by analyzing the general case of our models, we can get a sense of whether or not directly solving for solutions will yield solutions that are unique. We will analyze the method of solving for a solution in the circular convolution case to gain greater insight on whether or not we can expect at most or at least one answer. The guarantee of these unique solutions to our problem ensures that our answers are truly correct and not simply one of the many possible options, which would obviously lead to greater confidence in our answers.

2 Introduction

The procedure of accurately scanning or recognizing a barcode occasionally ends in failure, meaning the data acquired was faulty and not much reliable information can be extracted. Being a subsidiary of image processing itself, research and improvement in our ability to process grainy or distorted images to recover what was originally captured is applicable to a plethora of things from crime fighting to research itself. My main focus is the application of linear and abstract algebra based methods to analyze the uniqueness of solutions from synthetic images created through circularly convolving a barcode with a Gaussian kernel and then adding random noise. In using synthetic barcodes to do experiments, we will use the general model equation of:

$$f = k * u + n \tag{1}$$

where f is a matrix representing the corrupted barcode signal, vector n is synthetic noise, vector u is our original barcode, and vector k is the blur kernel we will use to convolve by. Another variance of this model equation we will use will be:

$$f = k \odot u + n \tag{2}$$

in particular to represent circular convolution.

With the way UPC-barcodes are made to store information, a regular signal can be minimally represented by a 95-dimensional vector. Therefore, in order keep the data and computation as simple as possible, we make the assumption that our u variable will always be at this minimal dimension of 95. With the conventions of circular convolution, it is also reasonable to assume that our blur kernel k will also be at most 95 dimensional, to match u. If k is a vector smaller than 95 dimensions, it must be padded with zeroes in the beginning and end to reach the 95-dimension length.

3 Circular Case

For the analysis of the case involving circular convolution, the bulk of our analysis will revolve around representing convolution between two vectors by an equivalent expression of matrix multiplication instead by converting k into a circulant matrix, K_k:

$$k = \begin{pmatrix} k_0 \\ k_1 \\ \vdots \\ k_n \end{pmatrix} \rightarrow K = \begin{pmatrix} k_0 & k_n & k_{n-1} & \dots & k_1 \\ k_1 & k_0 & k_n & \dots & k_2 \\ \vdots & \vdots & \vdots & & \vdots \\ k_n & k_{n-1} & k_{n-2} & \dots & k_0 \end{pmatrix}$$

with K_k, our model changes slightly to become:

$$f = K_k u + n \tag{3}$$

We begin by making the assumptions that k is a symmetric vector of dimension m. Also, the operation of circular convolution is shown to be commutative through use of Fourier transformations. Furthermore, we will denote a "flipped" vector with a large tilde over it, such as $\tilde{k}$. It is worth noting that the circulant matrix of a flipped vector, which we will denote with a subscript $K_{\tilde{k}}$, is not in general equal to a flipped circulant matrix of the original vector, K_k. Therefore:

$$k = \tilde{k}$$

$$K_{\tilde{k}} \neq \tilde{K}$$

Following a string of logic akin to one introduced in the paper by Dr. Lou[1], we begin with $f \odot \tilde{u}$ and start the derivation by substituting in $(k \odot u)$ for f, giving us:

$$= (k \odot u) \odot \tilde{u}$$

The assumption that k is symmetric and circular convolution is commutative justifies the next few steps:

$$(\tilde{k} \odot u) \odot \tilde{u} = (\tilde{k} \odot \tilde{u}) \odot u = S(\tilde{f}) \odot u$$

The derivation gives us the equality where $f \odot \tilde{u}$ is equal to a "slightly shifted" $\tilde{f}$ circularly convolved with our barcode u. From here we can move all terms to the same side to essentially get a homogenous linear system. Once we move both terms to the same side, we will convert all of the circular convolution operations into their corresponding matrix multiplication forms by circulant matrices to get:

$$F_{S(\tilde{f})}u - F_f(Eu) = 0$$

We choose represent $\tilde{u}$ by the product of vector u with an elementary matrix, E, that flips the order of its entries. We then factor our the vector u is get the basic for of a homogenous system:

$$(F_{S(\tilde{f})} - F_f E)u = 0 \tag{4}$$

Our preliminary tests in Matlab has yielded a constant rank of 47 for our system. Given that full-rank would be 95, since this is the dimension of u, all experiments do indeed point towards the nullity being greater than 0 and thus solutions to this system being non-unique.

We are using in-house code written for Matlab to produce a representative $1x95$ vector to be our synthesized barcodes. We have further tried the possibility of increasing the reference widths of our synthesized barcodes and changing the blurring sigma values of our blur kernel, but none ultimately have had any effect on the rank of our $(F_{S(\tilde{f})} - F_f E)$ system. This conclusion came independent of conditions as the use of stretching matrices to increase the number of equations to work with netted the same half dimensionality.

4 Conclusion

In the short period of research, we had managed to get a general idea for an answer to our question. With multiple cases pointed towards the non-uniqueness of solutions, we can either count this as a counter example to a uniqueness conjecture and conclude this research question over or move on to create a mathematical proof of non-uniqueness. Given more time, we would certain look at the possibility of constructing such a proof and then get the chance to scrutinize other methods of image recovery. We were already in the midst of analyzing a second method of image recovery that extensively used Z-transformations to work with polynomials and greatest common divisors- something we would definitely revisit.

References

[1] Jing Wang Ming Jiang Yifei Lou, Bin Zhang. Blind deconvolution for symmetric point-spread functions. 2005.