Mobile Capture of High-Resolution Data-Bearing Markings

Matthew Gaubatz, Hewlett-Packard Co., Seattle, WA, USA
Stephen Pollard, Hewlett-Packard Co., Bristol, UK
Robert Ulichney, Hewlett-Packard Co., Andover, MA, USA
Steven Simske, Hewlett-Packard Co., Ft. Collins, CO, USA

Abstract

Recent developments in data-bearing print include a number of technologies that enable high-capacity encoding of data via manipulation of high-resolution printed structure (halftones). While it has been shown that data encoded with these methods can be decoded from scanned images, there are challenges associated with achieving the same type of functionality on a mobile imaging device. The image must be captured at a distance that can resolve the required detail. At the same time, it is necessary to locate the desired region of interest quickly, and to determine if the region is in focus. Due to the designs of several common mobile phone APIs, in some applications it is necessary to achieve this result using frames captured at video rate/quality, and without direct control of an imaging device's focus system. Nonetheless, the proposed approach is capable of extracting information embedded in the halftone structure of 600 dpi prints with a mobile device.

Introduction

Recent developments in data-bearing print include a number of technologies that enable higher capacity encoding of data than traditional schemes (such as 1-D, DataMatrix, QR, MS Tag, etc.) via manipulation of halftones. Some examples are based on halftone dot orientation [1] or position [2] (see Fig. 1). One of the appeals of these approaches is that they offer greater graphic design flexibility in the creation of data-bearing print. While it has been shown that data encoded with these methods can be decoded from scanned images, there are challenges associated with achieving the same functionality on a mobile imaging device. This work describes these challenges and presents a system designed to overcome them, specifically to allow information to be gleaned from the halftone information.

The solution involves the combination of a fast object detector [3] and multi-scale alignment scheme [4]. The object detector has some distinct advantages when it is applied under the constraints of the problem, most notably that it decomposes the problem of detecting an object into stages that involve measuring the degree of the presence of simple piecewise-constant structures. Though this tool is capable of detecting complex objects in natural images, the fact that print contains *explicit* bi-tonal (i.e., piecewise-constant) structures makes it an appropriate choice for the solution. The approach is tested with several devices in conjunction with steganographic halftones, which require sub-pixel accurate alignment to enable correct decoding. It is shown that this strategy can achieve correct decoding on error-protected payloads embedded in prints that are captured with mobile imaging devices.

This paper is organized as follows. The following section discusses practical considerations that help define the problem. Next, the proposed solution is described, and test results are given in the last section.

Figure 1. *A QR code (left) and a design with d halftone structure (right).*

Constraints and Challenges

Candidate solutions must accommodate several considerations in order to be practical. First, they must provide imagery with enough visual acuity to resolve information in the high-resolution structures. Second, the solutions must be implementable on a range of devices.

Resolving the Target

One of the most difficult aspects of creating a mobile reader is that it must be capable of establishing content that is in focus. *Stegatones* [2] represent one high-resolution method of conveying information where the positions of halftone clustered dots are used to embed data in print. Typically, the halftones used for this purpose are rendered at 400-600 dpi, and in general, these markings pose few problems for decoding from captured scans. An example stegatone print is given in Fig. 2. To be properly resolved and interpreted, the captured print must usually exhibit a resolution that is at least as great as the resolution of the original rendering. Obviously, some devices will not be able to do so depending on the quality of the camera. Nonetheless, on devices that demonstrate the ability to resolve enough relevant details, the problem of capturing an in-focus image must be solved.

This issue is illustrated with the following example, based around an iPhone 4S. Let D represent the distance to a target in cm. The resolution of pixels captured by the camera is given by

$$\text{resolution (dpi)} = 1565 - 91.7\,D. \qquad (1)$$

Tests used to establish this relationship also indicate that the objects closer than 8 cm are not in focus. Similarly, objects captured beyond 11 cm will generally not have enough resolution

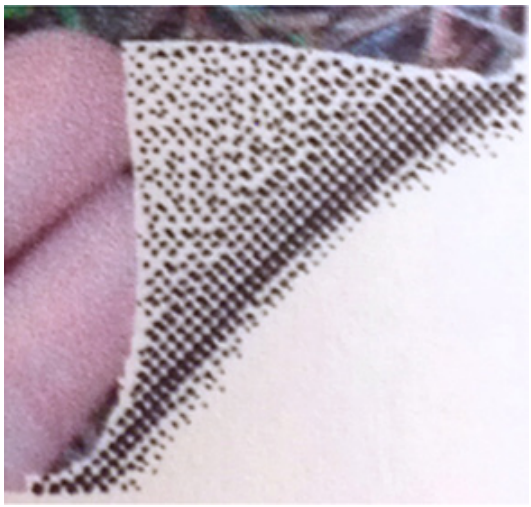

Figure 2. *An enlarged view of a captured data-bearing "flip-up" graphic; the capture and subsequent reading process is used to extract information from minute perturbations.*

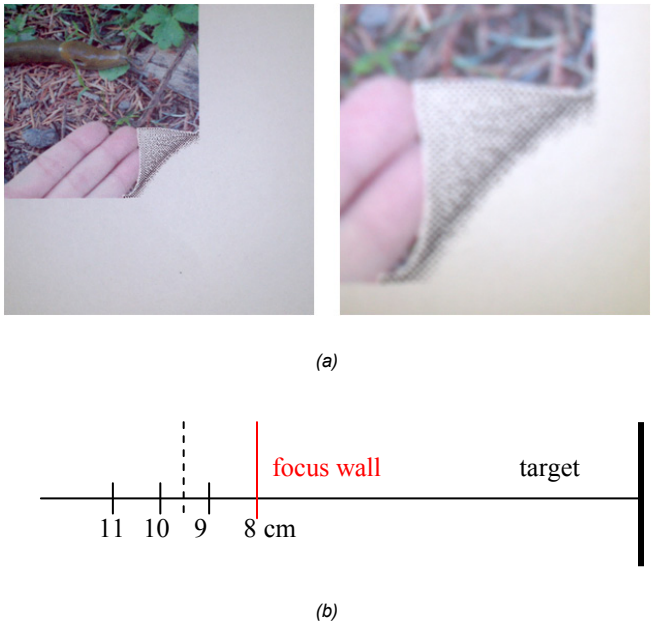

Figure 3. *(a) Examples of captured prints that are (1) in focus, but at too low a resolution for proper interpretation, and (2) close enough for decoding, but not in focus. (b)* ***D****istance-to-target window where a steganographic halftone, encoded at 600 dpi) can be captured in focus, and with enough visual resolution for decoding.*

to be properly interpreted. The absolute measurements discussed here will vary from device to device; the key point is that the distance-to-target window where a data-bearing print can be correctly interpreted can be relatively small. Fig. 3(b) shows data bearing print examples that were captured in-focus, but too far from the target for proper decoding, and close enough to include all the necessary detail needed for decoding, but out of focus; the iPhone window is illustrated in Fig. 3(b). A mobile reader must be set up to be able to determine the difference between these prints and an ideal candidate for decoding.

There are several methods that can be used to address the constraints mentioned above, two of which include (1) guiding the user through the process of capturing an image of the target (common, for instance, in mobile check-cashing tools [5]), and (2) determining the information via constant monitoring of a video stream. While the first option allows a greater degree of control from the algorithm design perspective, and for more efficient (explicit) focus estimation, it does not offer a user experience comparable to that achieved with, for example, QR codes (see Fig. 1). The ubiquity of QR codes in printed media has popularized a mechanism by which users interact with print. The goal is to achieve the same interaction with mobile device and a high resolution marking, ideally without guiding the user through a capture procedure. Furthermore, given the narrow window in which focus can be achieved, any user action to manually begin a capture process might actually distort the achieved focus at any given point. As a result, this work examines solutions that can be automated through monitoring streaming images.

Multi-Platform Implementation

While proof-of-concept implementations are a good first step towards developing a mobile reading procedure, ideally, a solution would be applicable to a class of devices, as opposed to just one. This constraint suggests limitations to general algorithm complexity, as well as possible focus analysis/estimation schemes.

First, a solution must be fast enough to function on a variety of different hardware platforms with different degrees of computational power, especially given expectations associated with QR reading applications. A number of current generation mobile devices have some (albeit limited) graphics processing capability, but at present, this functionality is not widely available, cross-platform option. Therefore, this work focuses on low-complexity algorithms for general processors that implement a front-end for reading applications.

Second, different devices offer different modes of operation, and control over those modes, to a programmer. For example, the Android API includes calls that specify focal distances, but not every phone implements them. Many phones do provide either a video mode that attempts to establish focus on a continual basis, or a method of invoking and auto-focus operation programmatically. Thus, by using either of these techniques, capture via continuous monitoring of image streams becomes possible. A simple way to provide a solution applicable to a range of devices is to perform much of the required computation off of the device, i.e., on a centralized server.

Distributed Object-Detection-Based Solution

The proposed solution involves a fast detection scheme to determine if a marking of interest is present. If so, further analysis is conducted. The goal is to reserve computational resources until they are needed, i.e., until a captured frame is likely to contain an example of a marking that will decode. Depending on the capabilities of a given device these steps can be performed either locally or remotely.

System Architecture

The proposed system combines front-end processing on the mobile device with other computation services to perform sub-pixel alignment of the image data, decode the image, and retrieve any digital information associated with it. This design is illustrated in Fig. 3. Some mobile devices may have the computational power to encapsulate the entire process, but at least at present, not all do. The video capture process operates in a loop that is decoupled from the print interpretation scheme. This scheme processes the frame that is available at the beginning of the analysis loop, and only proceeds if each of the first two stages are completed in succession. Ideally, when the marking of interest is out of the frame, no more subsequent processing occurs and the next frame is analyzed. If the marking of interest is detected, a test is conducted to determine if the region is of high enough quality to attempt alignment and decoding operations, both of which require more computation.

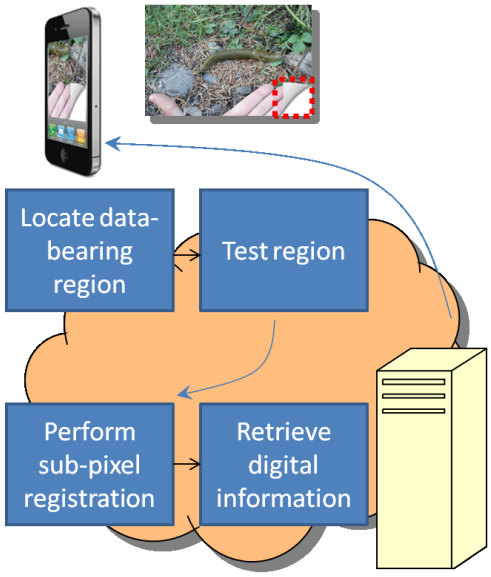

Figure 4. *Proposed reader architecture; on devices with enough processing power, more of the computationally intensive processing can take place in the front-end.*

Part of the reason for this particular design is that some of the alternative solutions still have issues to be addressed. If a camera is in continuous focus mode, it is possible to establish estimates of focus quality by tracking a number of measurements (such as band-pass energy) as a function of time. The issue with this type of approach is that it introduces lag, and furthermore is more likely to succeed after a calibration step, which may not be possible in all cases. Since back-end processing can be much faster than what is achievable on a mobile device, and estimates of decoding accuracy offer a device-independent, functional measure of captured marking quality, the proposed system attempts to decode each frame that passes the second stage.

Reader Design

A series of simple feature detectors, motivated by the Viola-Jones detection system [3], are used to implement the first stage of the system. This framework has been shown to be very effective at detecting faces real-time in video streams. In the context of the current problem, it is used to detect objects that are generally simpler to separate from background images. Printed markings are piecewise constant signals, which are quite different from most photographic content. Furthermore, piece-wise constant signals are convenient to separate from *each other* using the Viola-Jones framework due to the fact that the features are computed as linear combinations of piecewise-constant functions. Therefore, this framework can separate data-bearing markings from non-printed-objects and printed objects alike with relative ease. Finally, a multi-scale detector constructed with this approach can be configured to only search for markings of sizes given by the constraints on distance-to-target window.

Some of these ideas are quantified with the following. The classifier generation technique used herein [3] automatically determines simple features, which are computable in constant time, to determine whether a given object is present in a candidate image window by exhaustively evaluating training data. A set of 25 images of printed media and 25 images of general photographic content were collected. Single-feature object detectors were trained using true positive samples that were generated from affine perturbations of several example target objects, and false positive samples that were randomly selected from the 50 images. In particular, 625 true positives (representing all combinations of 5 scaling operations, 5 angle perturbations, and 25 translations, i.e., 5 x-direction translations x 5 y-direction translations) were generated from each specified example target. A total of 5000 false positives, 100 from each image, were generated at random. True positive samples are illustrated in Fig. 5.

Table 1: Single Haar-like feature [3] classification results achieved on training set, as a function of example targets.

Target image	*False positive rate*	*Hit rate*	*Source*
flip-up	*0.0046*	*1.0*	*digital*
smoke	*0.0052*	*1.0*	*digital*
logo	*0.0060*	*1.0*	*photo*
slug	*0.0016*	*1.0*	*photo*
wheel	*0.0038*	*1.0*	*scan*
boat	*0.0060*	*1.0*	*scan*

Table 1 illustrates the classification performances achieved when applying the detectors on the training set. Note that these simple filters are able to eliminate over ninety-nine percent of the false positives in a single pass, and that the success rate is not very dependent on the source of the example. Because this tool can narrow the search space for target objects so efficiently, it is an attractive candidate for the first reader stage.

While the object detection scheme is fast, the method used for sub-pixel alignment is very general. This multi-scale alignment scheme [4] establishes a transform that is used to orient the marking of interest in stages, each of which involves a comparison between band-pass filtered versions of a reference halftone and a captured marking. Because this method is robust to slight degradations in focus, it enables attempted decoding of captured frames that vary quite a bit in quality. It can be applied on any type of marking, and does not require any offline training.

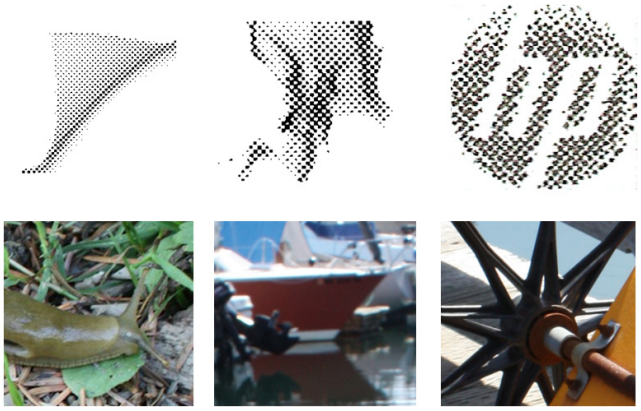

Figure 5. *Example targets used to test the reader design. Clockwise, from top-left: flip-up, smoke, logo, wheel, boat, and slug.*

Once aligned, a candidate captured data-bearing halftone can be decoded relatively quickly. The result can then be passed back to the client application. As the capture and recovery process is not error-free, error correction coding (ECC) forms a key part of the stegatone transmissions scheme; ECC, if it was used, is applied at this time as well. In this application, the ECC is used as another indicator that a legitimate message was decoded.

Results and Discussion

The proposed approach has been applied to data captured by a Motorola Atrix, and to an extent, an iPhone 4S. The Atrix represents what is achievable on one of the more computationally powerful devices, and the iPhone demonstrates what is possible with very good optical capabilities.

End-to-End Decoding Performance

Table 2 features decoding performance for the *logo* image, printed and imaged in several difference ways. This particular printed marking carries a 16 byte payload with approximately 4x redundancy (the raw payload size is 612 bits). The raw and message recovery rates correspond to the percent of bits correctly decoded in the raw stream, and in the ECC protected message. Note that while it is not possible to correctly interpret a message encoded in the marking rendered at 600 dpi on the Atrix, it is possible with an iPhone. This difference is due to the higher resolution of the iPhone, and the fact that it can be closer to the target while maintaining focus. Though decoding fails in this case, this result can be predicted based on the *predicted block error rate*, which represents the percent of code words in the bit stream associated with a reported ECC error. Generally, this number should be small for correct decoding to occur. In the case of the 600 dpi print imaged by the Atrix, the majority of the ECC bits detected errors, implying that it is improbable the marking will be interpreted correctly. While it would be convenient to have client-side techniques to predict/prevent such an outcome, this mechanism provides another layer of robustness for the reader.

Future Work: Scalability

Though the proposed approach demonstrated the ability to locate a marking accurately, one of the issues to be addressed in the future is scalability of the solution. While the Viola-Jones detection scheme is very fast during online computation, the offline training procedure can be extensive, requiring tens if not hundreds of hours of computation time. This issue is further compounded by the need to create appropriate training sets. Thus, a more scalable solution is desirable. In limit of massively parallel computation, the proposed algorithm can be deployed with relative ease. Several improvements on the original Viola-Jones framework have been developed that improve computation required by an order of magnitude [6,7]. Alternate implementations with feature-based techniques that in principle, could be simultaneously used to detect and register the content [8,9] are also of interest, as faster implementations emerge.

Table 2: Reader performance on *logo* image, as a function of different devices and render resolutions.

Property	*Mobile device*			
	Atrix	*Atrix*	*iPhone*	*iPhone*
Size (in^2)	*½ x ½.*	*¼ x ¼*	*½ x ½.*	*¼ x ¼*
Resolution (dpi)	*400*	*600*	*400*	*600*
Message payload (bytes)	*16*	*16*	*16*	*16*
Raw recovery rate (%)	*99.8*	*56.3*	*94.8.*	*92.8*
Message recovery rate (%)	*100*	*61*	*100*	*100*
Predicted block error rate (%)	*6.5*	*73*	*24*	*18*

References

[1] O. Bulan and G. Sharma, "High Capacity Color Barcodes: Per Channel Data Encoding via Orientation Modulation in Elliptical Dot Arrays," IEEE Transactions on Image Processing, 20, 5 (2011).

[2] R. Ulichney, M. Gaubatz, and S. Simske, "Encoding Information in Clustered-Dot Halftones", Proc. IS&T NIP26, pgs. 602-605 (2010).

[3] P. Viola and M. Jones, "Robust Real-Time Object Detection," International Journal of Computer Vision, 57, 2 (2001).

[4] J-Y. Bouguet, "Pyramid Implementation of Lucas Kanade Feature Tracker: Description of the algorithm", Intel Corporation, Microprocessor Research Lab, OpenCV Documents (1999).

[5] https://www.chase.com/online/services/check-deposit.htm.

[6] J. Wu, C. Brubaker, M. Mullin and J. Rehg, "Fast Asymmetric Learning for Cascade Face Detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, 30, 3, (2008).

[7] M.-T. Pham and T.-J. Cham, "Fast Training and Selection of Haar features using Statistics in Boosting-based Face Detection," in Proc. IEEE ICCV, pgs. 2-7 (2007).

[8] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," International Journal of Computer Vision, 60, 2, (2004).

[9] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, "SURF: Speeded Up Robust Features", Computer Vision and Image Understanding, 110, 3 (2008).

Author Biography

Matthew Gaubatz is a research scientist at HP Labs. He received his Ph.D. in Electrical Engineering from Cornell in 2006. His current focus is on functional imaging methods that support anti-counterfeiting research.