An Optimization Model for Aesthetic Two-Dimensional Barcodes

Fang Chengfang, Zhang Chunwang, *Ee-Chien Chang*
School of Computing
National University of Singapore

NUS
National University of Singapore

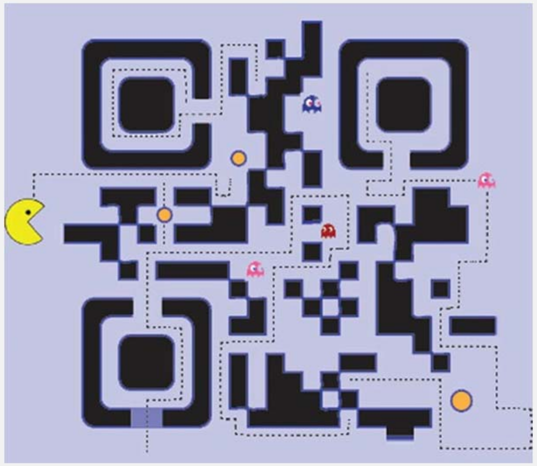
1. Motivations

Aesthetic Barcodes

- 2D barcodes visually resemble random dots. We want to make them visually interesting to encourage usages, and/or to piggyback visual information on the QR code.

- Examples of manually designed barcodes.

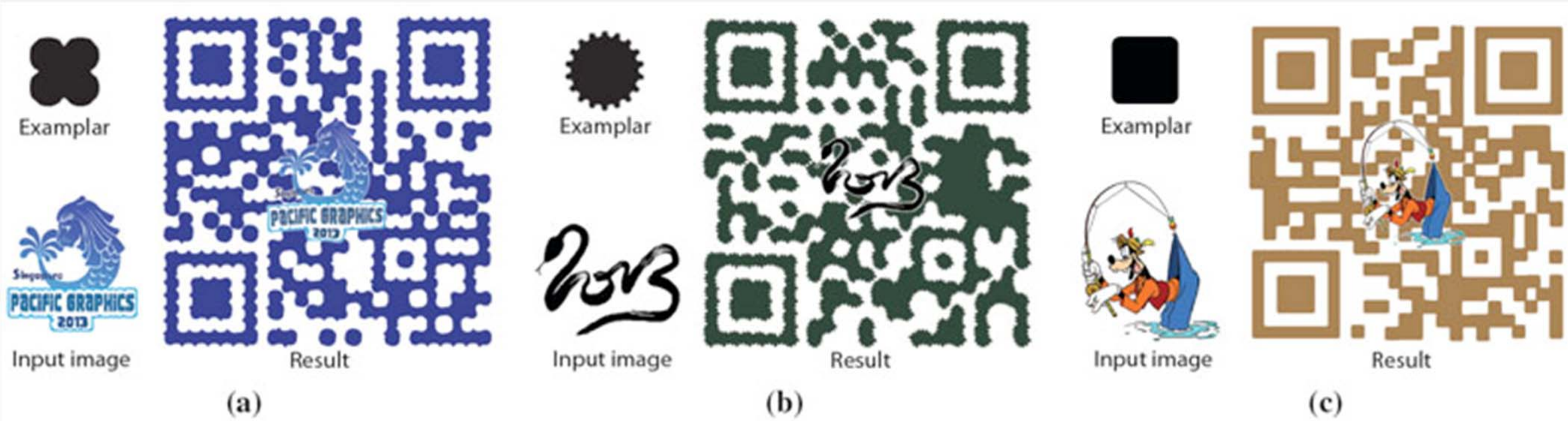
From **15 Beautiful and Creative QR Codes.** (http://mashable.com/2011/07/23/creative-qr-codes/)



Automated Generation

- Given a message and an image, automatically generates the aesthetic barcode.
- Manually designed barcodes are expensive.
- Able to readily or interactively generate barcodes that carrying different messages.

Existing automated method

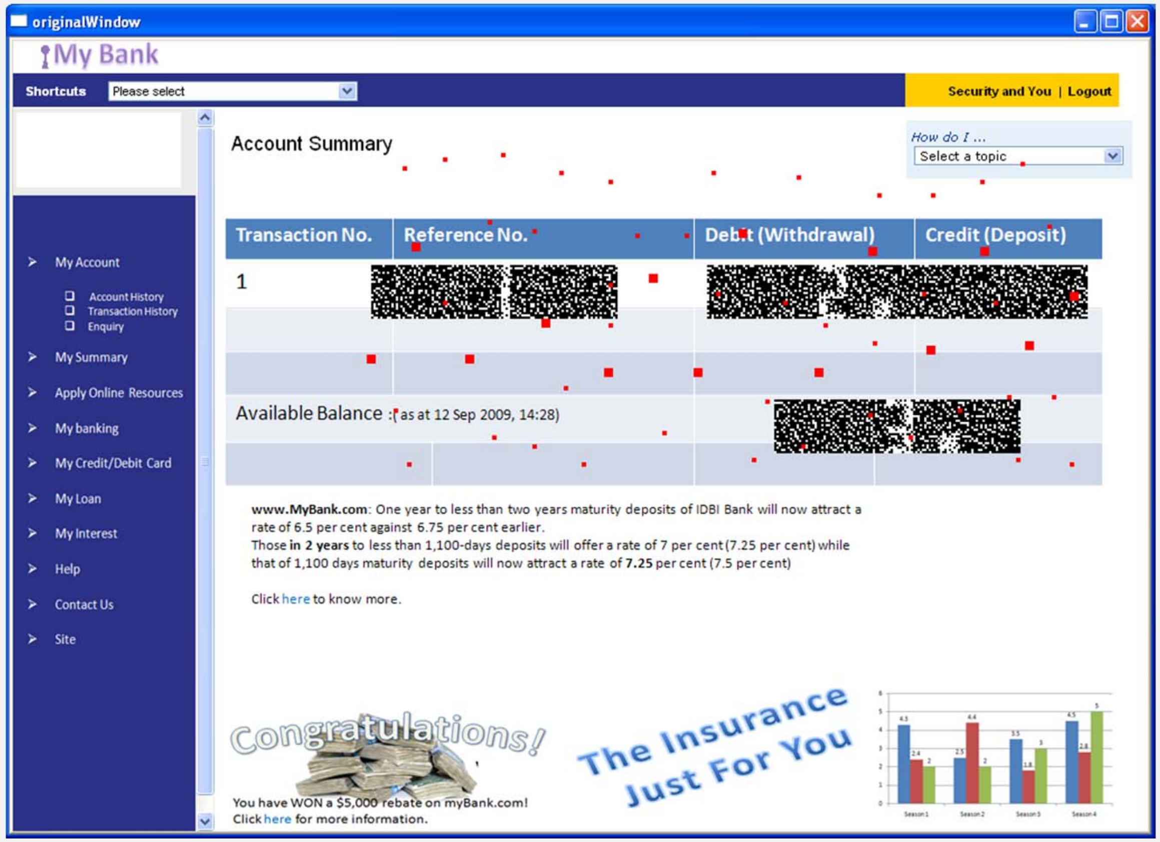


from http://graphics.csie.ntu.edu.tw/~forestking/research/PG13-QRCode/

Zachi et al
Visually Significant QR codes: Image blending and statistical analysis

Message: hhtp://www.united.com

Using “half-toning”



Chengfang Fang and Ee-Chien Chang, *Securing Interactive Sessions Using Mobile Device through Visual Channel and Visual Inspection*, Annual Computer Security Applications Conference (ACSAC), 2010.

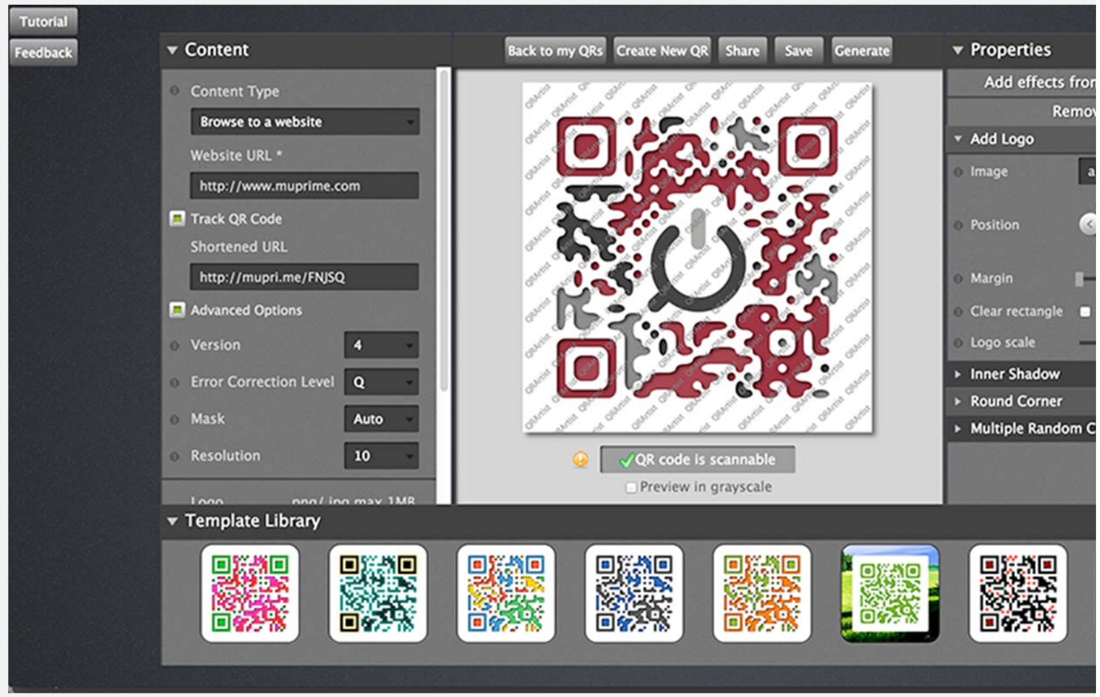
Commercial systems

Visualead
http://www.visualead.com/

QRArtistic
http://qrartist.net/

(image created using the generator in www.visualead.com)

Message:
http://VQR.MX/fxm5



(image obtained from http://qrartist.net/qrartist-product)

Message:
http://mupri.me/FNJSQ

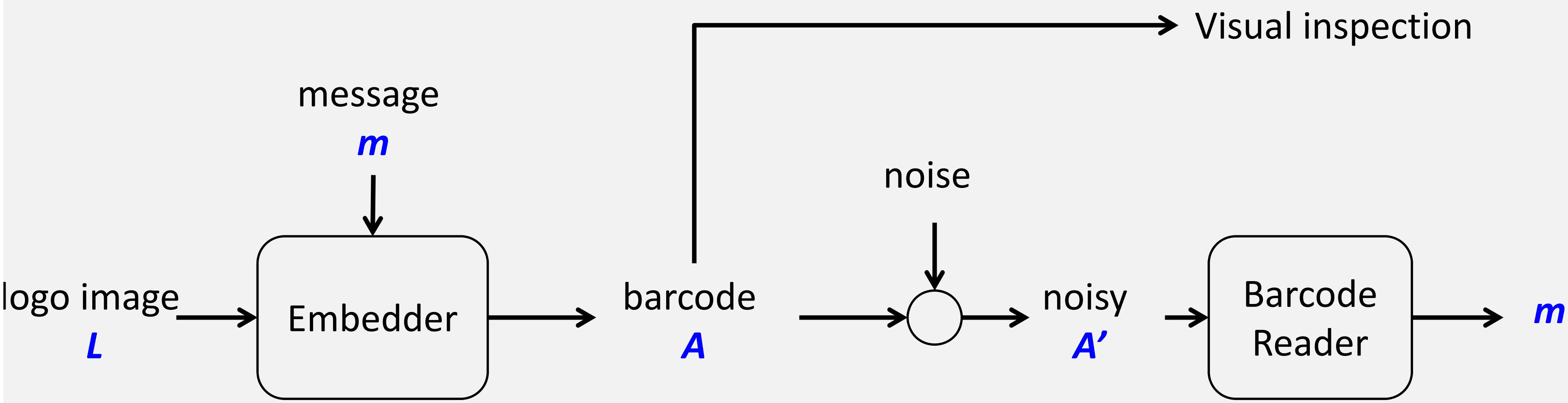
Examples of our barcodes

Message:
http://www.comp.NUs.EDU.sg/~cHAnGEc/#12345678901234567890

Message:
http://www.comp.NUS.edU.SG/~cHAnGec/#12345678901234567890

2. Problem Formulation

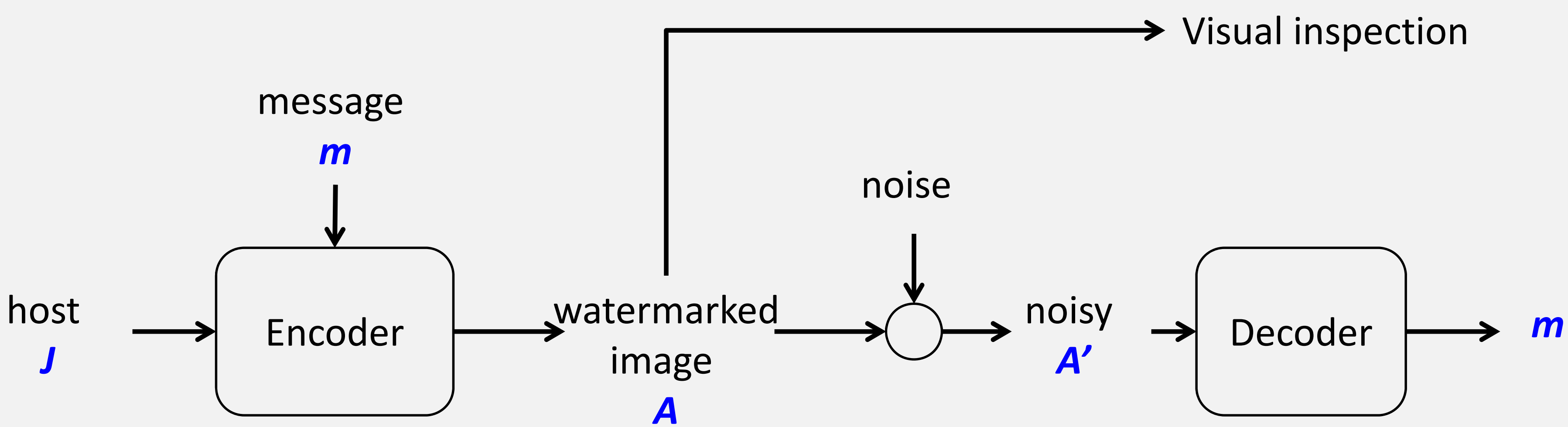
Proposed Model



Given a Barcode Reader, we want to design an embedder that maximizes the visual similarity between ***A*** and ***L***, and yet ***m*** can be correctly decoded with high probability.

The Barcode Reader is given and cannot be modified.

Digital Watermarking



Here, the decoder can be modified by us.

(*i.e. we want to design an encoder and decoder that maximizes visual similarity, and yet* ***m*** *can be correctly decoded with high probability.*)

Main components in proposed model

- Visual similarity.

$D_w(A,L)$: Qualities the visual distance between the given image L and the barcode A, where w is a weight function that can be derived from J or specified by the user.

- Successful decoding of the message.

F(A,m): Estimate the expected additional bit-error incurred by the barcode A compare to the "default" barcode in encoding the message m.

2.1. Problem Formulation: Visual Distance function

Visual distance function D_w (A,L)

- In this paper, we adopt weighted Euclidean distance.

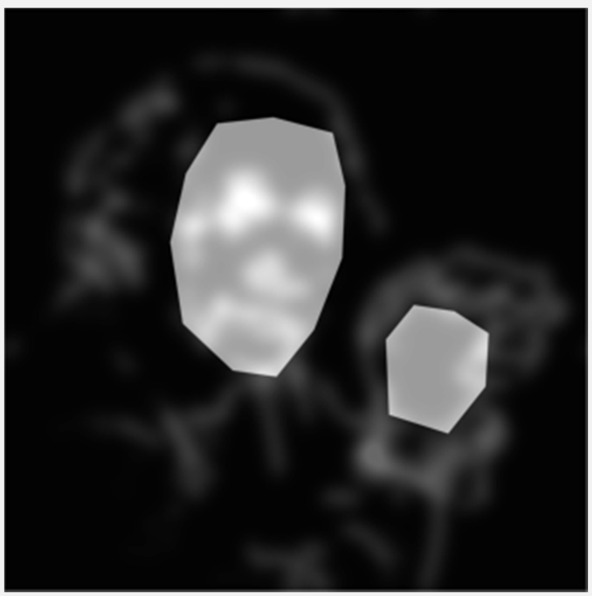
$$D_w (A,L) = \sum_i w_i (a_i - \ell_i)^2$$

Our solution can handle other quadratic functions. For e.g. letting the user to specific the weight of edges in L.

$$D'_w (A, L) = \sum_i w_i \{edge(A,i) - edge(L,i)\}^2$$

where edge(A,*i*) is a linear combination of the pixel values surrounding the *i*-th pixel.

- The weight function can be manually specified, derived from salient features, or a combination of both.



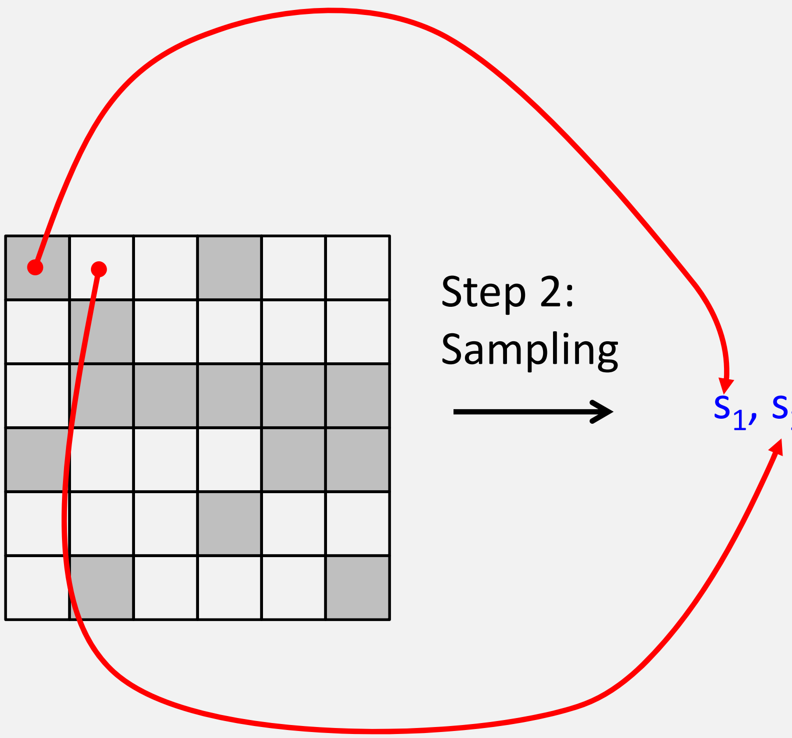
2.2. Problem Formulation: Expected error

Estimating the expected error

- We investigate the process carried out by a typical barcode reader.

- Note that a barcode is composed by a grid of square “modules”. Each module consists of a number of pixels in the digitized image.

Steps carried out by a typical reader



Step 2: Sampling

$s_1, s_2, \ldots, s_{36}$

Step 3: threshold to 0,1

$b_1, b_2, \ldots, b_{36}$

Error correction

message m

Steps carried out by a typical reader

- **Step 1:**

Captures the barcode as an image. Perform synchronisation (i.e. geometric alignment).

- **Step 2:**

Using a sampling function (i.e. weighted average) to obtain the intensity of each module

$$\boldsymbol{s} = \langle s_1, s_2, \ldots, s_n \rangle$$

- **Step 3:**

Threshold to a bits sequence, where each bit correspond to one module

$$\boldsymbol{b} = \langle b_1, b_2, \ldots, b_n \rangle$$

- **Step 4:**

Apply error correction code to decode the message m from $\boldsymbol{b}$.

Step 2, sampling function

- Typical reader use a sampling function (i.e. weighted average) to obtain the intensity of a module.

- Examples of sampling functions: Gaussian-like 2D function, or average of a few pixels surrounding the center.

- For each *i*-th module in the barcode *A*, we maps the sampled real value s_i to an estimated probability that the corresponding bit would be "thresholded" to the wrong bit value. In current implementation, they are mapped to either 0.5, 0, or 1.

- Formulate F(A,m) to quality the expected number of bits in the sequence **b** (sampled from A) that are wrongly read.

3.1. Problem Formulation: Optimization

Optimization problem

Given a message m, a logo image L and a weight function w and a threshold e, find a barcode A that minimizes

$$D_w(A, L)$$

subjected to the constrain

$$F(A, m) < e.$$

III. Solving the optimization problem

Two phases.

Phase 1:

Find the intended intensity for each module that minimize visual distance, and yet the expected error is lower than the threshold. *(Dynamic Programming)*

Phase 2:

For each module, find the sub-image that minimizes the visual distance, and yet its sampled intensity is as determined in phase 1. *(Least square-fit for weighted Euclidean distance, or Quadratic Programming for general quadratic function.)*

III. Applying to QR Code

- Higher visual weight on “control points”.

- Exploit the error correction structure of QR code.

- Exploit error correction in the application layer. (For e.g. URL is case-insensitive. We search for the combination of capitalisation the minimize the visual distance.)

IV. Experiments and Analysis

Experiment setup

- On Toy Barcode, a simple barcode design.
- On QR code, using the reader “QR Barcode Scanner” on Andriod phone, Android tablet, and “QR Code Scanner” on iPhone and iPad, with the barcodes rendered on LCD monitor and printed in hardcopies (laser printer).
- Please refer to the paper for detailed results on robustness.

Inspecting the generated barcode

Different mappings of probability

p=20, t=5

t=15

p=20, t=25

t=45

IV. Conclusions

Conclusion

- We proposed a principled approach, and pointed out a key different from the well-known watermarking model.
- By inspecting typical steps carried out by a typical barcode reader, we formulate the process as an optimization problem that can be efficiently solved using dynamic programming + quadratic programming.
- Adopted the techniques to the popular QR codes.