Wearable barcode scanning

Advancements in code localization, motion blur compensation, and gesture control

Gábor Sörös
Doctoral examination
ETH Zurich
May 3, 2016
Linking the physical and the digital

atoms

objects

bits

data

services

01011101010
11010110100
01001011010
11001110001

01011101010
11010110100
01001011010
11001110001

01011101010
11010110100
01001011010
11001110001
Visual codes are everywhere
Wearable barcode scanning

Barcode scanners
- are expensive
- are used by only few people
- use proprietary protocols

Smartphones, tablets, watches, glasses
- are always with us
- have cameras, sensors, intuitive UI
- are easily programmable

Ubiquitous wearable scanners allow us to access information on every physical object
Challenges

- no laser for localization
- (multiple) small codes
- defocus and motion blur
- limited input capabilities
Research goals

- Make **wearable barcode scanning** an attractive alternative of traditional laser scanning
- by compensating the shortcomings, and adding **new features**
- by leveraging the advanced **computing** and **sensing** capabilities of the wearables
Contributions

Part I
Fast and robust localization of visual tags
MUM’13, ICASSP’14

Part II
Fast and robust blur compensation for scanners
WSCG’15, ISWC’15

Part III
Fast and robust gesture control for wearables
BSN’14, UIST’14, CHI’15
Fast and robust code localization

goals: invariant to size, orientation, blur, symbology
Observations

- 1D barcodes contain lots of edges. Blur deletes many of them.
- 2D barcodes contain lots of corners. Blur smears corners but they still remain corners.
- Codes are almost always black and white. Blur mixes black and white to gray.

Detect areas with edges and/or corners & low saturation in HSV color space.
Joint 1D and 2D barcode localization for smartphones

1D

2D
Live localization on the mobile GPU
Results

Our method

- can localize visual codes of various symbologies
- with performance like the state of the art
- without assumptions on code size, code orientation, or code position, while it is more robust to blur
- is portable to GPU and a wide range of devices
Multiple codes

1D sensitive to blur
2D works well in both cases
Extension to blurry 1D codes

Low $S_1$ and $S_2$
Rectangle detection in the saturation channel
Fast and robust code localization allows:

- scanning multiple codes simultaneously
- scanning visual codes from further away
- scanning blurry codes in the whole image
Contributions

Part I
Fast and robust localization of visual tags
MUM’13, ICASSP’14

Part II
Fast and robust blur compensation for scanners
WSCG’15, ISWC’15

Part III
Fast and robust gesture control for wearables
BSN’14, UIST’14, CHI’15
Motion blur compensation

motion blur makes the codes unreadable

we recover the information from motion-blurred QR codes

our input  our output

gabor.soros@inf.ethz.ch
Basics of blurry image formation

**uniform blur model**

\[ B = I \ast k + n \]

- **sharp scene**  
  \[ I \]
- **blurry scene**  
  \[ I \ast k \]
- **observed image**  
  \[ B = I \ast k + n \]

**convolution with a blur kernel**  
\[ k \]

**adding camera noise**  
\[ n \]
Blur removal problem

deconvolution:
\[ B = ? \ast k + n \]
blind deconvolution:
\[ B = ? \ast ? + n \]
Blind deconvolution for QR scanning?

Existing blind deconvolution algorithms

- are slow even on PC
- are tuned to natural images
- usually fail on QR codes (structure very different!)

![Image of QR codes and deconvolution results](image-url)
Observations for deblurring QR codes

- blur can be estimated from the many QR edges
  - but we need to suppress the small structures

- QR codes do not need to look good for decoding
  - in contrast to photographs, where restoration quality counts, our main concern is speed

- QR codes include error correction / checksum
  - the algorithm can stop when the checksum is correct
  - false decoding is practically impossible
  - only partially restored codes might be decoded too
Restoration-recognition loop

Blind deconvolution via energy minimization

\[
\arg\min_{I,k} \|B - k \ast I\| + \lambda_I p_I(I) + \lambda_k p_k(k)
\]

We follow a common recipe for blind deconvolution:

- **alternate** between solving for \( I \) and solving for \( k \)
- **suppress noise** and **boost edges**: enforce QR properties
- try to **decode** at every iteration
- repeat on several **scales**
experiments (synthetic blur)

Quality is on par with the state of the art, and a magnitude faster.
experiments (real blur)
Live deblurring on a smartphone

Main Activity

Camera view

Search window

Input

Estimated image

Kernel

Estimated kernel

Output Image
Can we make it even faster?

additional clues:

- the blur is 'encoded' in the image of point light sources
- wearables have inertial sensors
- rotational motion blur is dominant – use gyroscopes
- reconstruct the camera motion, render the blur kernel
Rendering blur kernels for initialization

Rotational blur depends on the position in the image.
Patch-wise restoration

We can initialize the restoration loop with the rendered kernels
Fast and robust blur removal allows:

- scanning in low lighting
- scanning moving codes
- and tiny or distant codes (super resolution)
Contributions

Part I
Fast and robust localization of visual tags
MUM’13, ICASSP’14

Part II
Fast and robust blur compensation for scanners
WSCG’15, ISWC’15

Part III
Fast and robust gesture control for wearables
BSN’14, UIST’14, CHI’15
Codes for interaction with smart objects

- Rohs 2005
- Ballagas 2006
- Heun 2013a
- Ballagas 2006
- Heun 2013b
- Mayer 2012
- Mayer 2014
- Heun 2013a
- Mayer 2014
- Chan 2015
Outsourcing user interfaces

The smartphone is becoming a universal interaction device.

How about other wearables?
Outsourcing user interfaces

cross-device automatic GUI generation: user interface beaming

Joint work with Simon Mayer
Gesture recognition on wearables

Joint work with Jie Song, Fabrizio Pece, Otmar Hilliges
Live gesture recognition on mobile devices
Gesture classification as pixel labeling
Pixel labeling with a decision tree

$F_0(w,v)$:

$F_2(w,v)$:
Pixel labeling with a decision forest

1. pooling over trees: this pixel is 'red'

2. pooling over all pixels: this gesture is 'red'
Pixel labeling with multi-stage decision forests

Coarse Depth Classification
- close
- middle
- far

Shape Classification
- pinch
- palm
- point
- splayed

Part Classification
Enabling 3D interaction

Coarse Depth Classification
- close
- middle
- far

Shape Classification
- pinch
- palm
- point
- splayed

Depth Regression
- 145 mm
- 211 mm
- 282 mm
- 323 mm
Gestures + depth for 3D interaction
Fast and robust gesture recognition allows:

- natural input to wearables
- easy control for scanners
- universal interaction with smart objects (through user interface outsourcing)
Conclusions

In the world of binary images, generally very difficult computer vision problems like …

... can have fast and robust solutions even on resource-constrained wearable devices.
Conclusions

Our solutions

- are pushing forward the state of the art in terms of **accuracy**, **robustness**, and **speed**
- can help to make wearable barcode scanning a **promising alternative** to traditional barcode scanning
- will potentially make wearables the essential tools for **bridging** the gap between the **physical** and the **digital** world.
Thank you!