A Fast Eavesdropping Attack Against Touchscreens

Federico Maggi, Alberto Volpatto,
Simone Gasparini, Giacomo Boracchi, Stefano Zanero

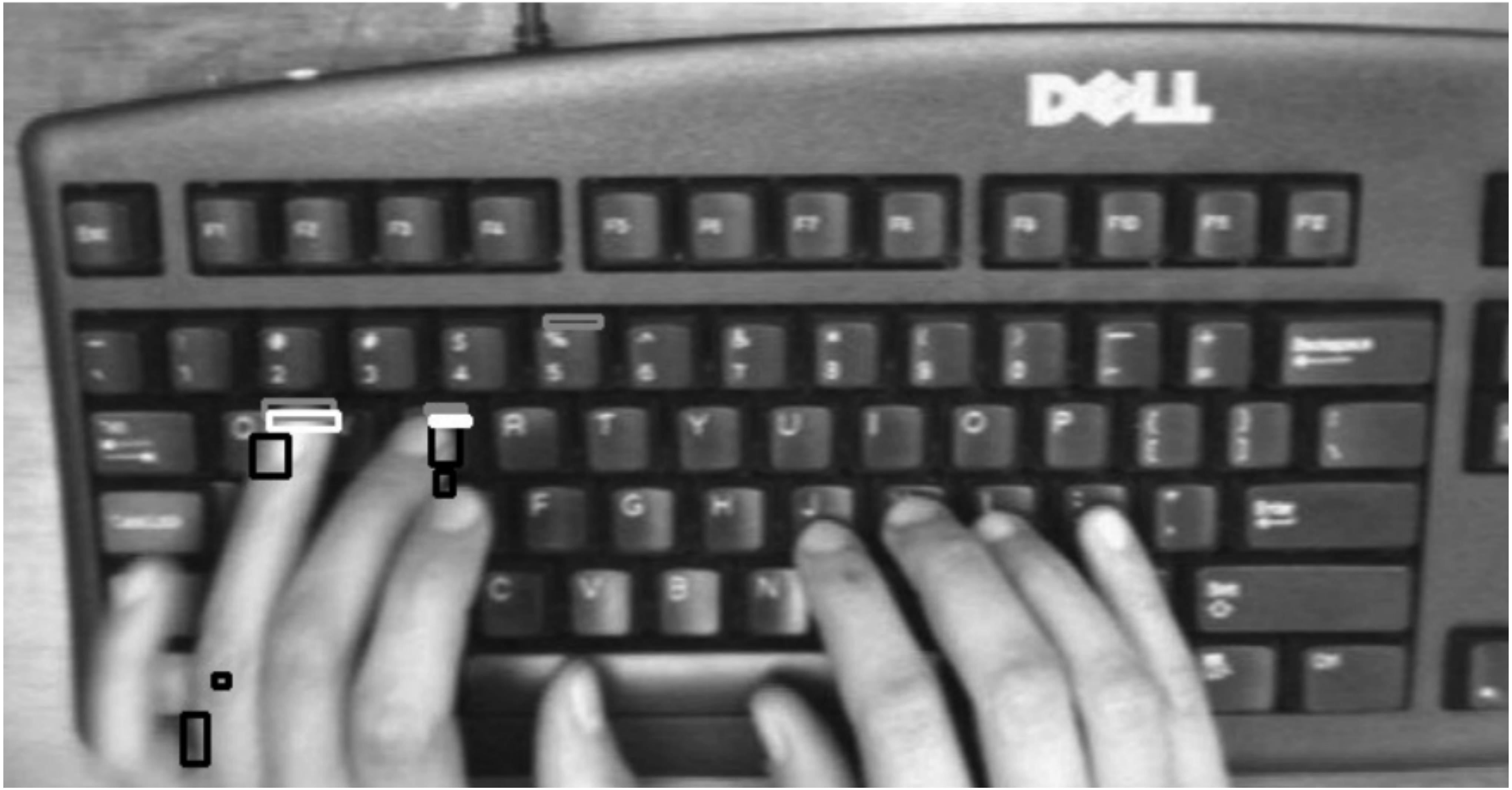
How sensitive data is compromised

- Direct attacks
  - Well-known in both literature and industry
  - Very active research community
- **Other types of attacks**
  - Social engineering attacks
  - Side-channel attacks
  - Difficult to mitigate (if not through awareness)

Side-channel Attacks

- Less known yet very effective
- Digital side-channels
  - Example: decrypting SSL through wifi LAN sniffing
- **Physical-world observation**
  - Direct observation
    - Shoulder surfing
  - Indirect observation
    - Sound emanations
    - Reflections
    - Magnetic radiations
    - Desk surface vibrations

Physical-world Observation



Automated Shoulder Surfing

- First attempt of **automatic** shoulder surfing
- Recovery of long texts

Ubiquitous Touchscreen Mobiles

- **2010** survey on 2,252 US citizens
  - 72% use a mobile phone for **texting**
  - 30% use a mobile phone for **instant messaging**
  - 38% use a mobile phone for Web **browsing**
- (1970) **touchscreen** technology was invented
  - 2010: **5 billion** US dollars market
  - 159% market **grow** rate
  - Q3 2010: 417 million of touchscreen devices sold

Automated Shoulder Surfing

- Non-automated
  - not interesting
  - time consuming
- Automated
  - Is it feasible?
  - Mobile context poses several constraints


New Message
To:
Cc/Bcc:
Subject:
Q W E R T Y U I O P
A S D F G H J K L
Z X C V B N M
space
return

Mobile Settings Constraints

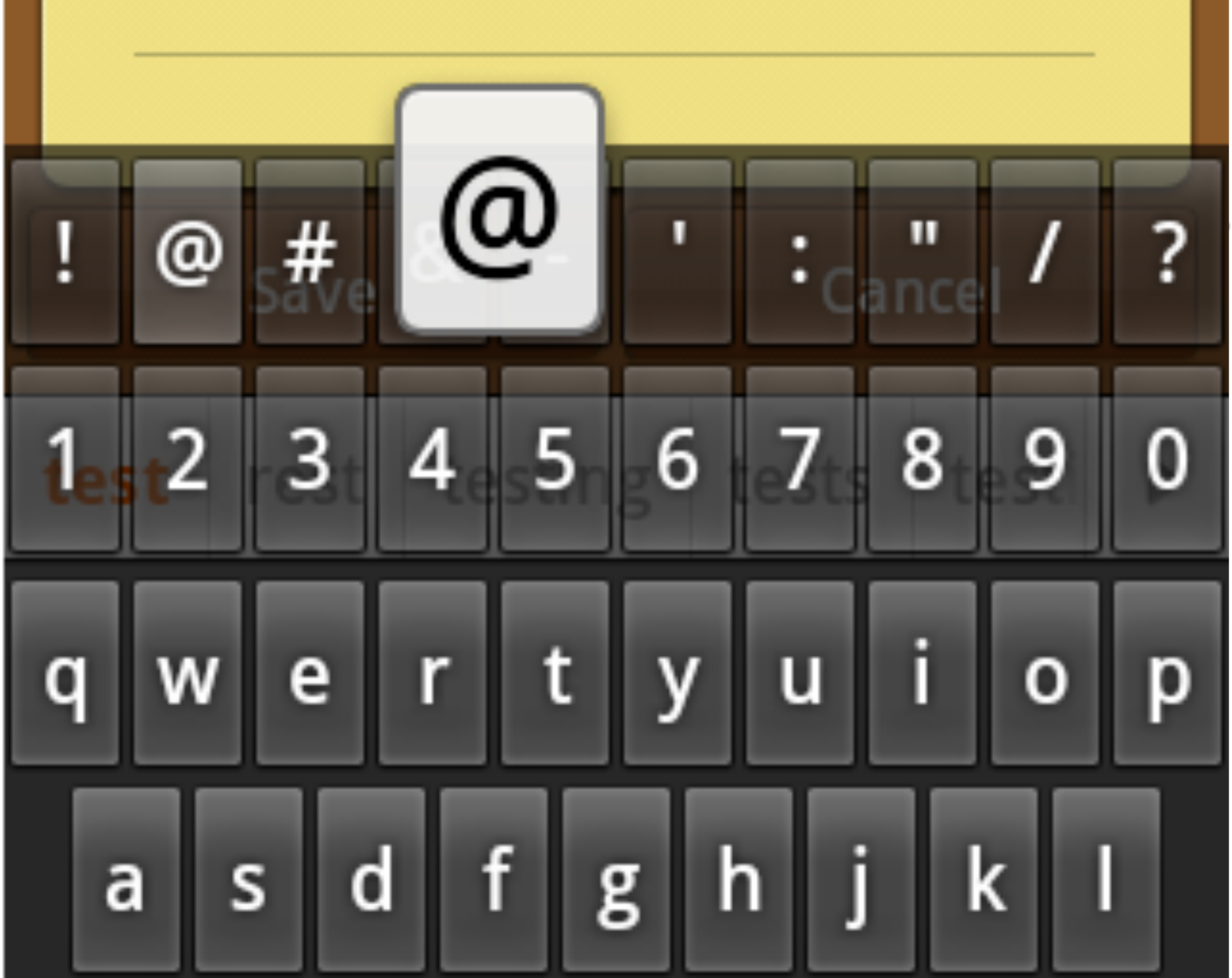
- Moving target
- Fixed observation point not always feasible
- Very small keyboards
- No visibility of pressed keys
- No visible key occlusions

Touchscreen to the rescue

- Lack of tactile feedback
- Early soft keyboards were hard to use
- UI engineers came up with **usable keyboards**

3 * + ( )
Q Y/ U I O P
6 # - "
A H J K L
9 0 ; :
N M , .
& ! Ctrl
? 中


AT&T
9:42 AM
Cancel
Re: Jenny's Birthday
Send
To: Rob Tucker
Cc:
Subject: Re: Jenny's Birthday
Great pi
pick
I
Q W E R T Y U
A S D F G H J
Z X C V B N M
.?123
space
return


@
! @ # ' : " / ?
Save
Cancel
1 2 3 4 5 6 7 8 9 0
test rest testing tests
q w e r t y u i o p
a s d f g h j k l

Usability vs Security

- Old dilemma
- More secure, less easy to use
- Example: Google's 2-step authentication
  - Very secure
  - Very unusable
    - Wait for the verification code every time you do email
- Apply also in this context
  - Feedback-less touchscreen keyboards
    - hard to type on
  - Feedback-rich keyboard keyboards
    - easy to type on
    - eyes follow the feedback naturally during typing


AT&T
9:42 AM
Send
Cancel
Re: Jenny's Birthday
To:
Rob Tucker
Cc:
Subject:
Re: Jenny's Birthday
Great pi
pick
I
Q W E R T Y U
A S D F G H J
Z X C V B N M
return
space
.?123


New Message

Our approach

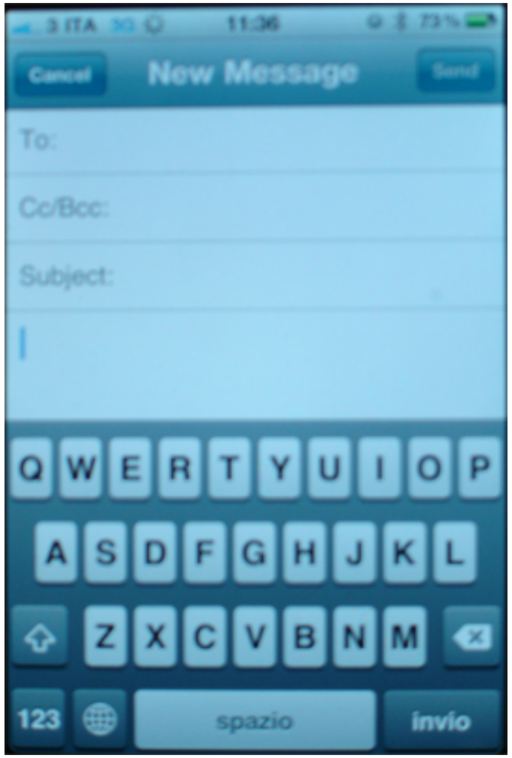
Simple Threat Model

- **Requirement 1**
  - iPhone-like visual feedback mechanism
- **Requirement 2**
  - Template of the target screen known in advance

Requirement 1 is often satisfied

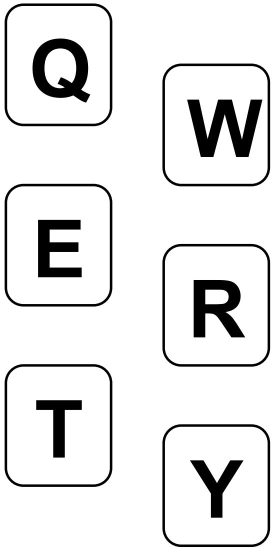
Requirement 2 is very easy to satisfy

SCREEN TEMPLATE



(screenshot)

KEY TEMPLATES



(synthetic, hi-res)

MAGNIFIED LAYOUT

(x,y-coordinates)

Outline of the Approach

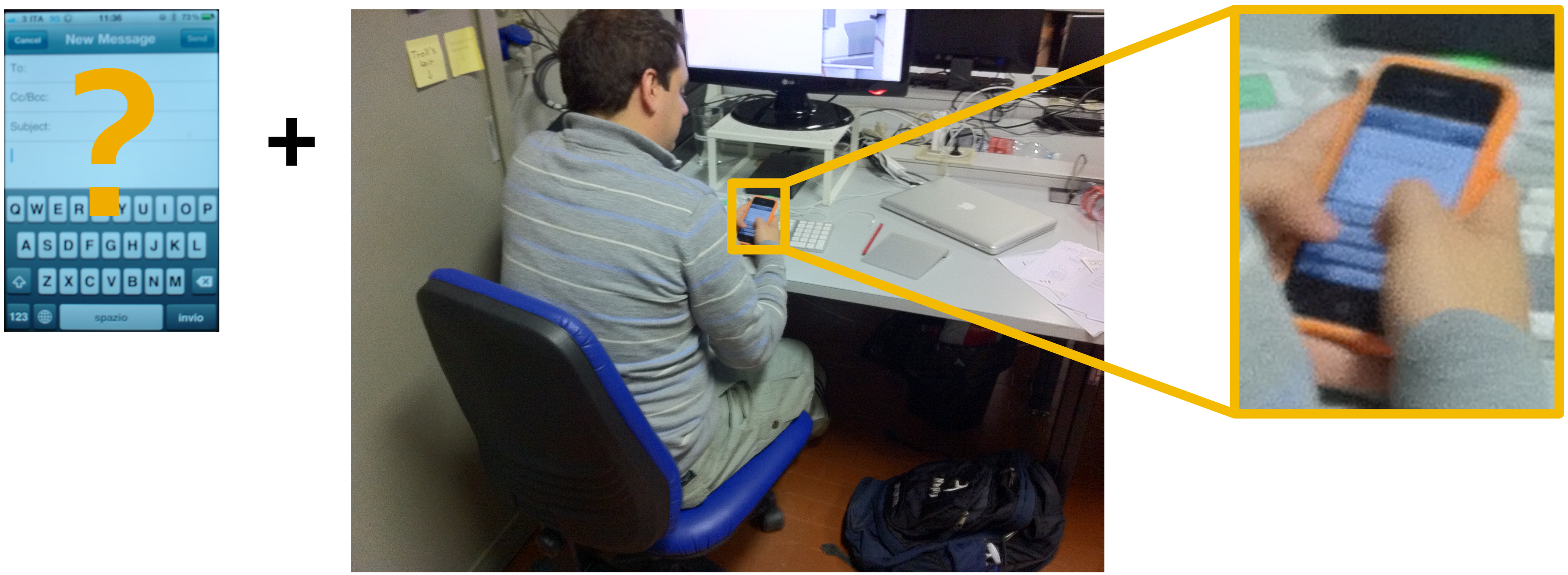
- **Phase 1**
  - Screen detection and rectification
- **Phase 2**
  - Magnified key detection
- **Phase 3**
  - Keystroke sequence reconstruction

Phase 1

- **Input**
  - Image depicting the current scene (current frame)
- **Output**
  - Synthetic image of the rectified, cropped screen
- **Procedure**
  - Screen detection
  - Screen rectification

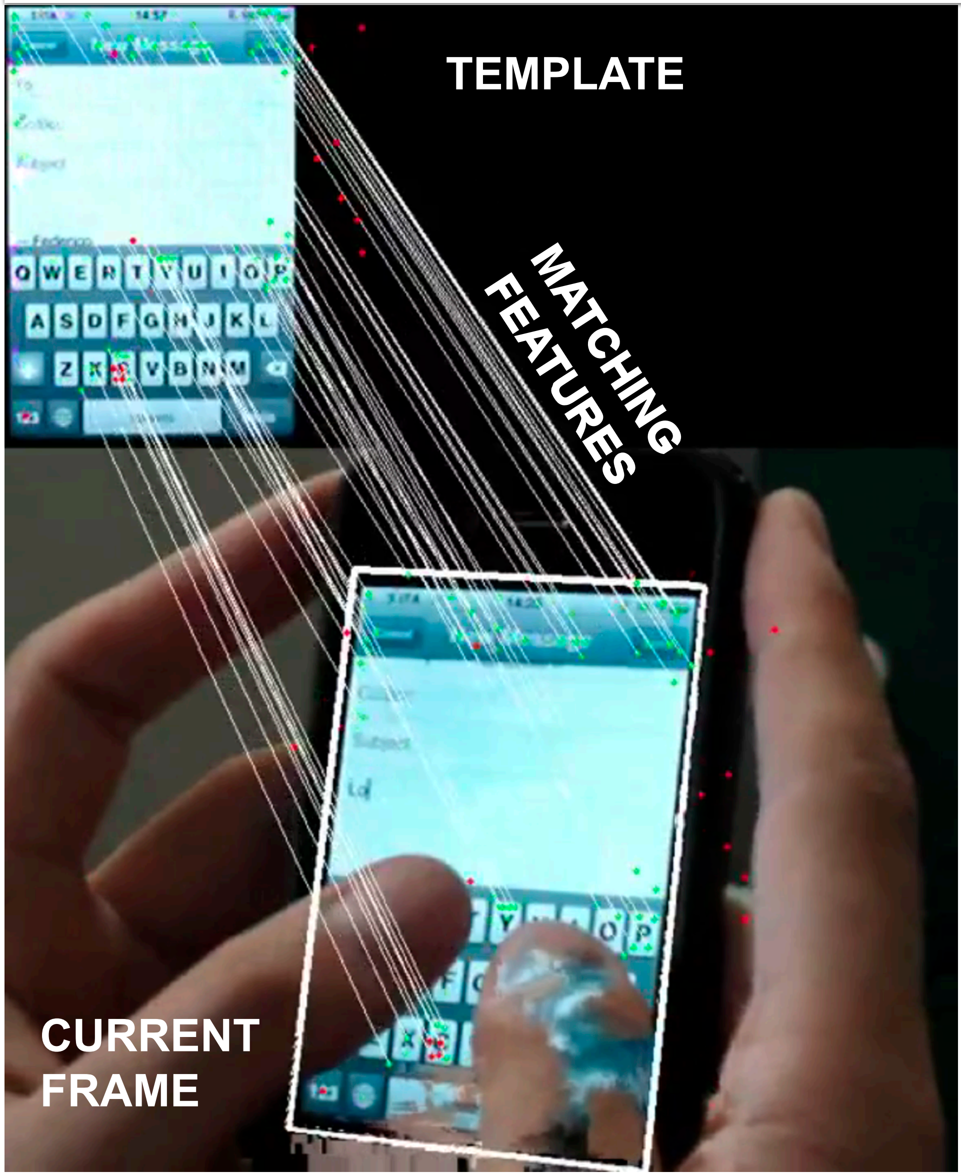
Screen Detection

- The current frame is searched for the screen template (Requirement 1)



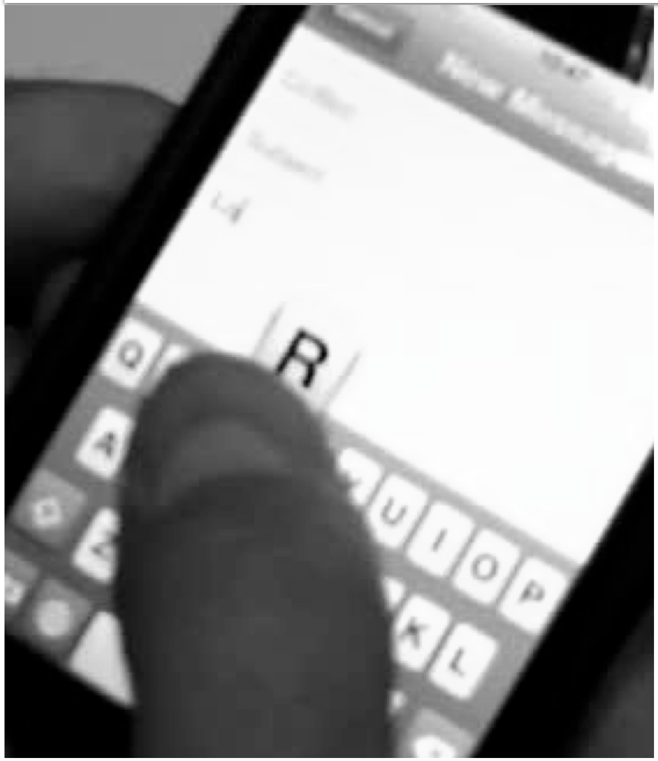
SCREEN TEMPLATE **CURRENT FRAME** **MATCHING PATCH**

Screen Detection via Template Matching

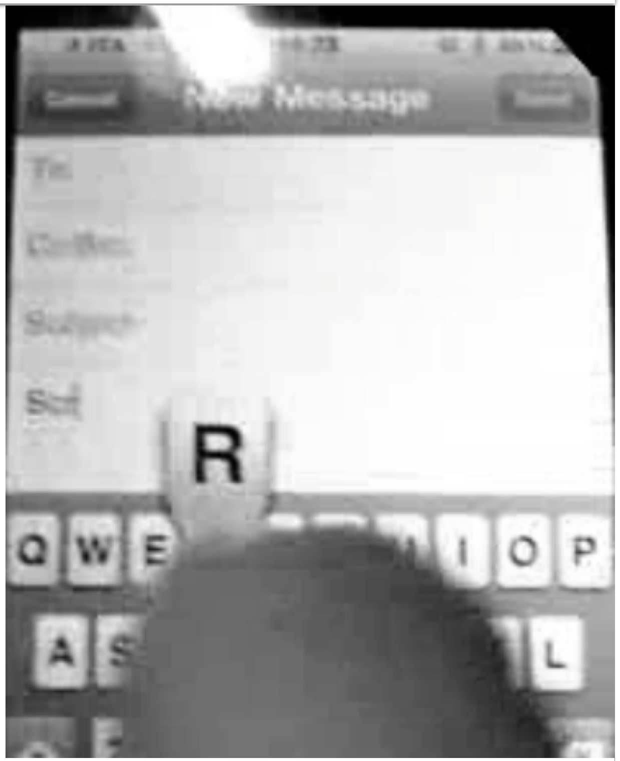


- **SURF** features
  - Edges
  - Corners
- Invariant to:
  - Rotation
  - Scale
  - Skew
  - Occlusions
- **Homography** estimation

Screen Rectification via Homography



- Estimate during screen detection
- Successfull matches improve matches in subsequent frames



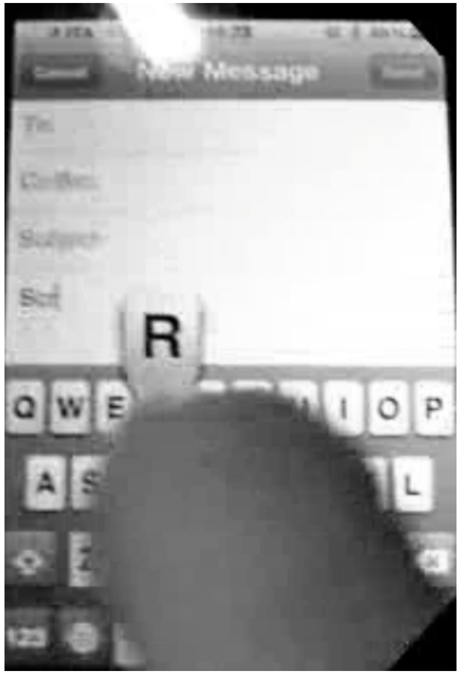
CURRENT FRAME

RECTIFIED FRAME

Phase 2

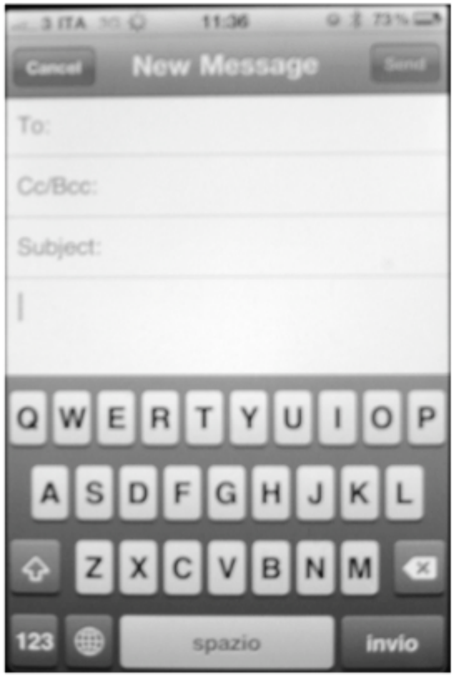
- **Input**
  - Image of the rectified screen
- **Output**
  - Areas where magnified keys appeared
- **Procedure**
  - Background subtraction

Pixelwise Background Subtraction



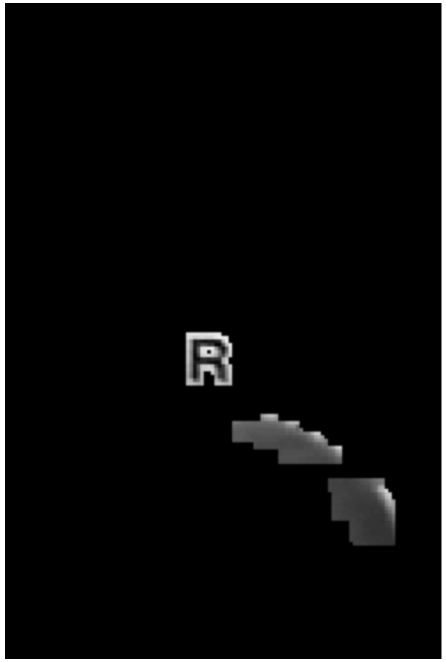
CURRENT FRAME

-



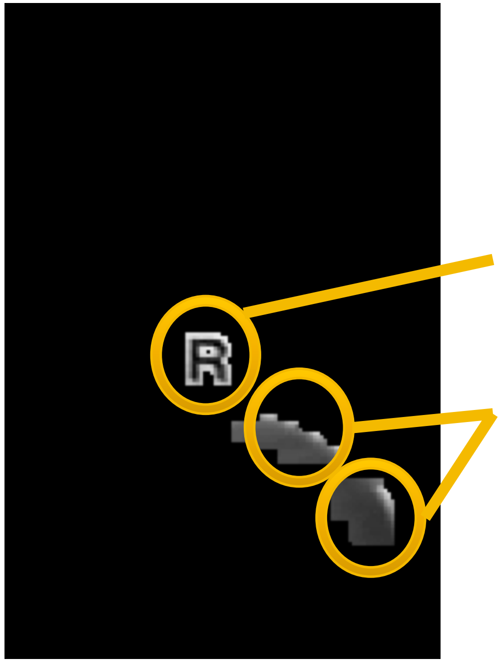
SCREEN TEMPLATE

=



FOREGROUND

Spurious output



HIGHLIGHTED KEY (MAGNIFIED-KEY CANDIDATE)

OTHER FOREGROUND ELEMENTS (NOISE)

FOREGROUND

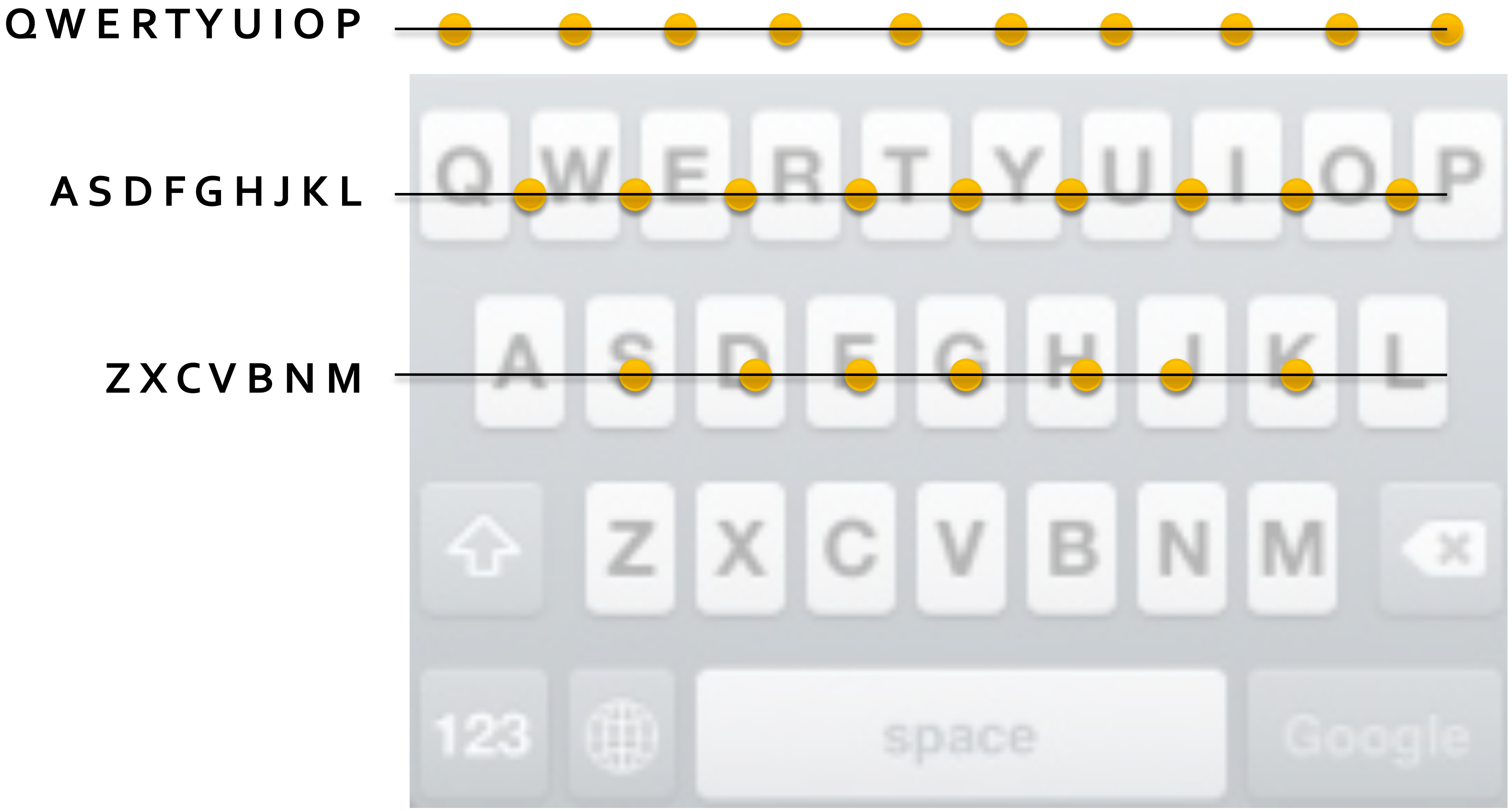
Phase 3

- **Input**
  - Magnified-key candidates
- **Output**
  - Sequence of typed symbols
- **Procedure**
  - Approximate neighbors lookup
  - Best matching key identification
  - Fast pruning
  - Key sequence analysis

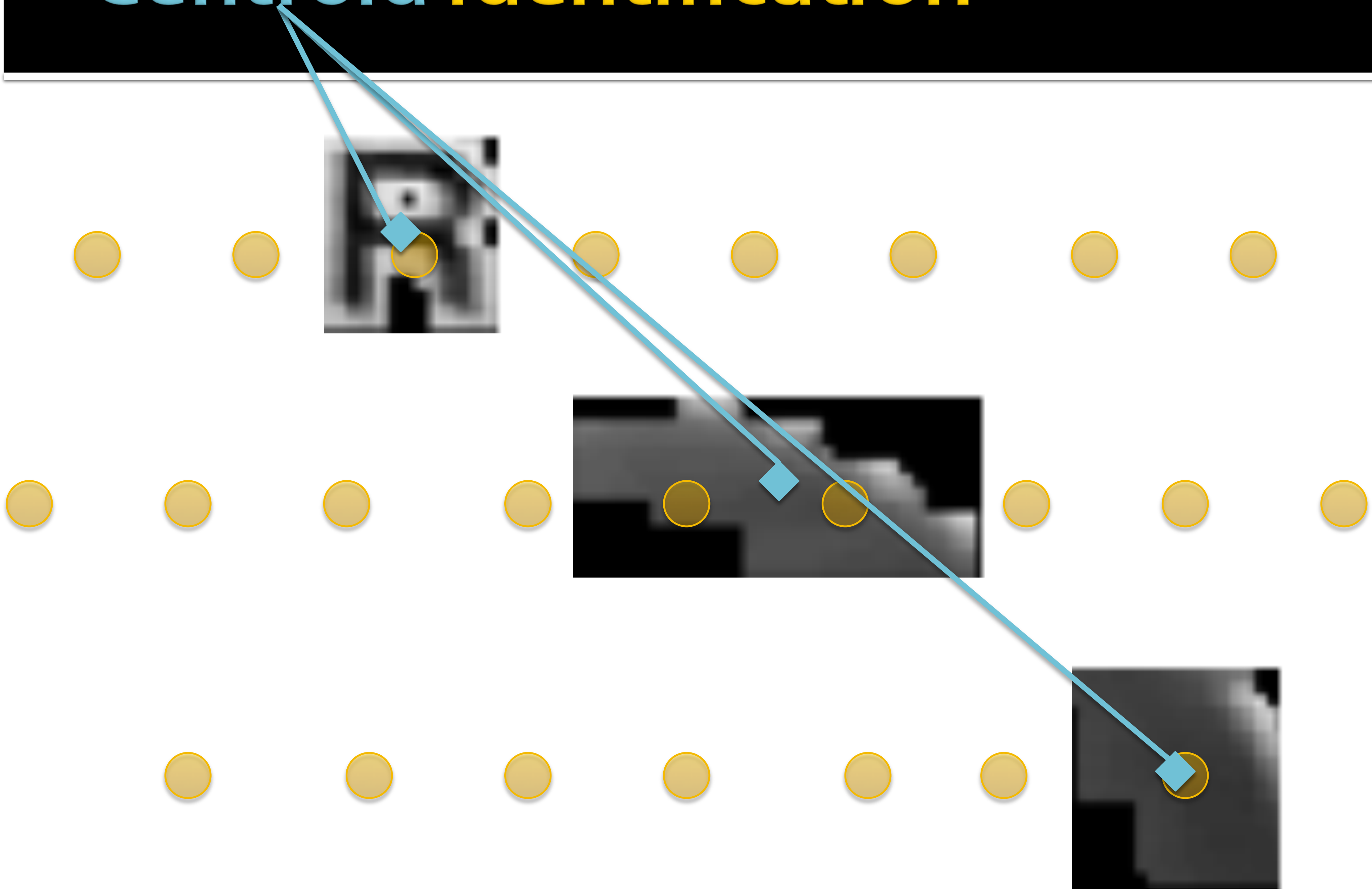
Approximate Neighbor Lookup

- Known keyboard layout (Requirement 2)
- Centroid identification
- Match centroids with keyboard layout

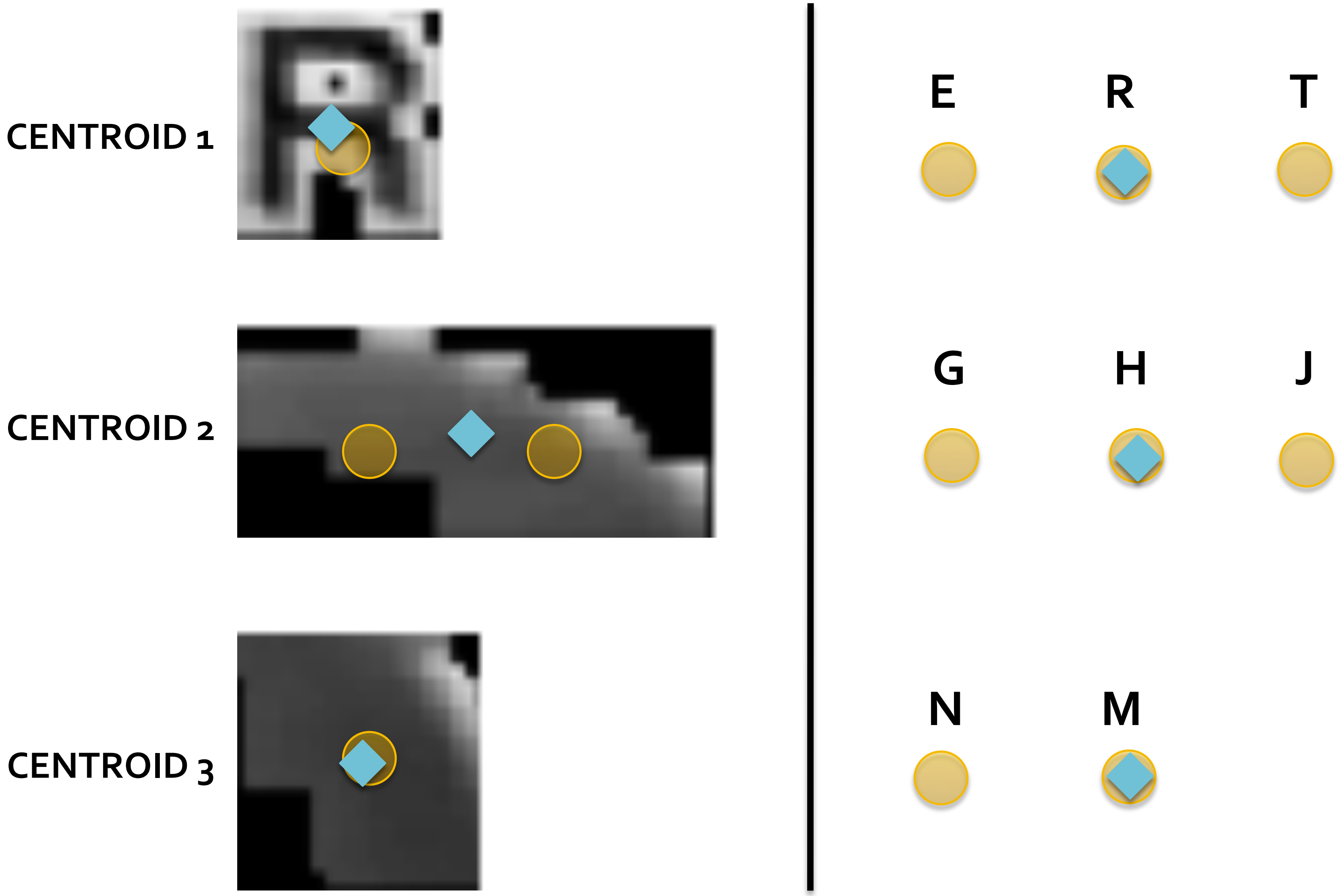
Known keyboard layout



Centroid identification

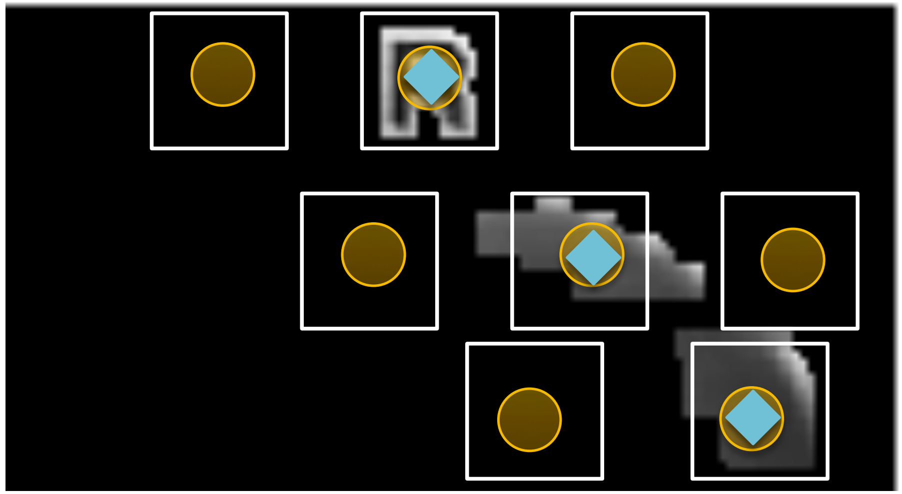


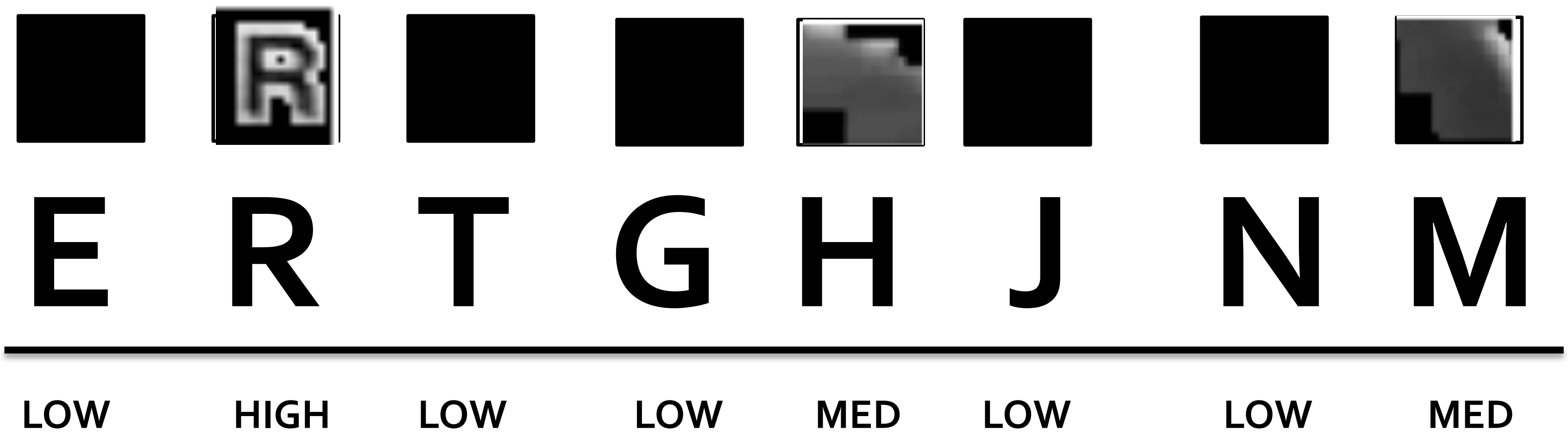
Match centroids with layout



Key similarity

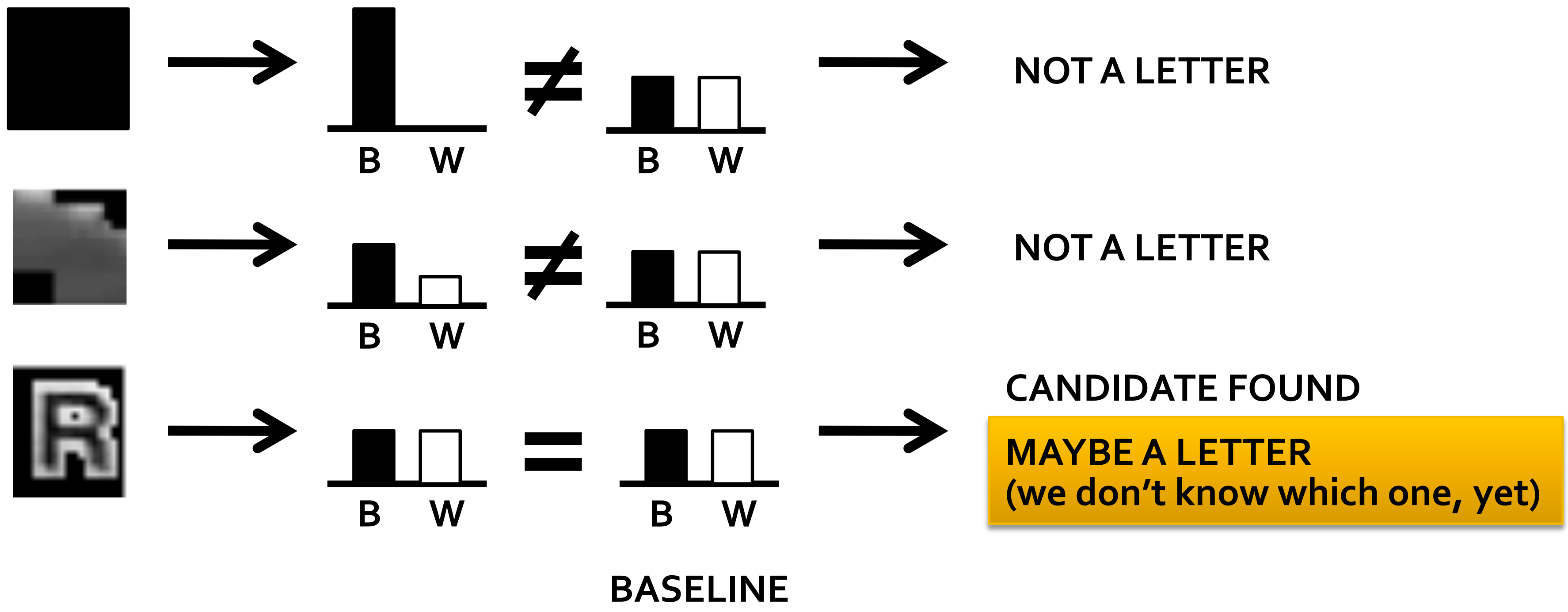
- Region of interest
- Key template (Req. 2)





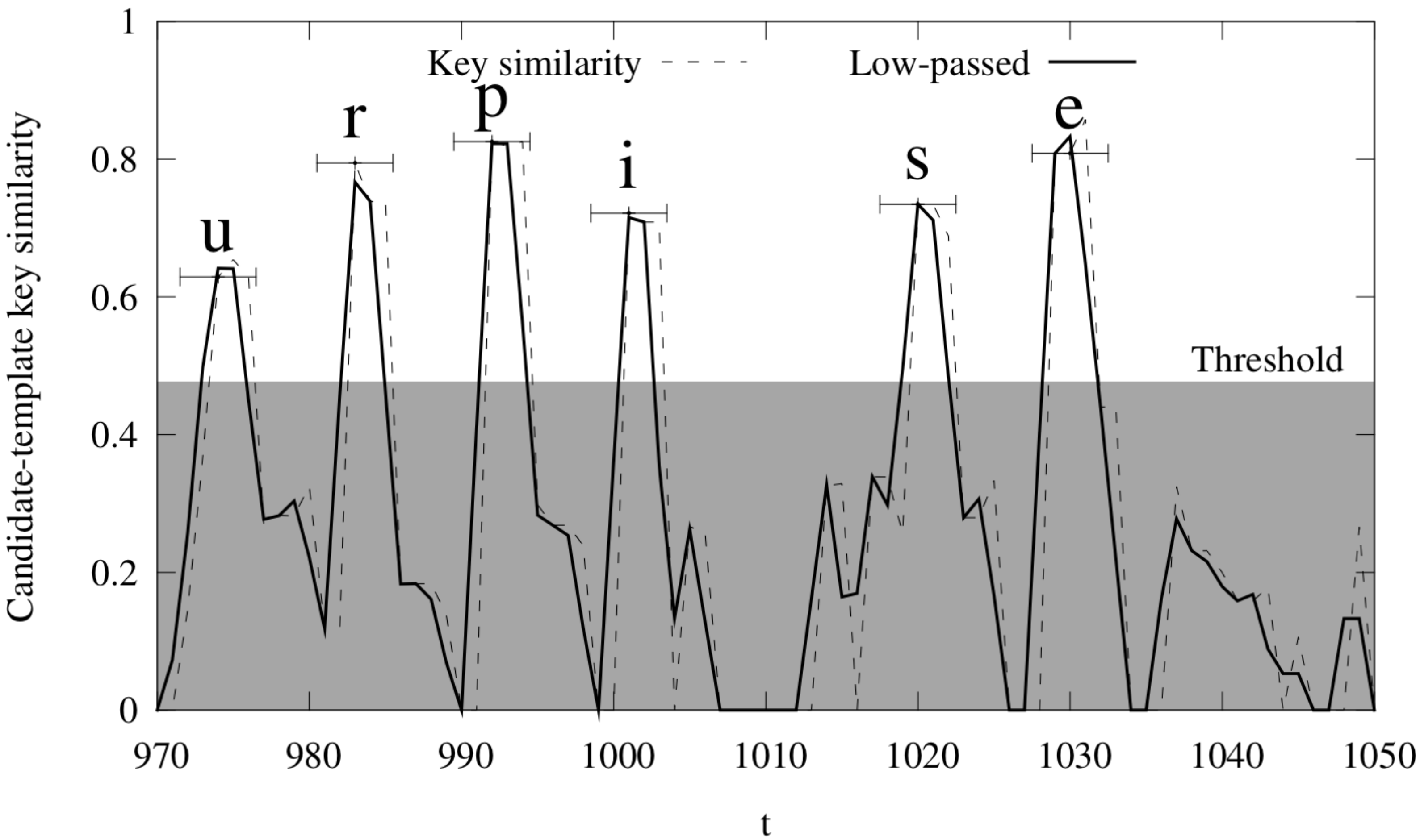
Fast Pruning

- Computing the key similarity is **expensive**
- Black-white distribution of the ROI
- %B/W-**heuristic** is way faster



Key Sequence Analysis

- Find maxima of the key similarity function



Implementation Details

- **Phase 1**
  - C++
  - OpenCV
- **Phase 2-3**
  - Matlab
  - Compiled into C
- **Threshold estimation**
  - Confidence interval (mean, variance)
  - Video samples collected in “no typing” conditions

DEMO

http://www.youtube.com/watch?v=aPuS8kNI30U

http://www.youtube.com/watch?v=t9BxB3dOoKQ

Experimental Evaluation

- Types of text
  - Context-free
  - Context-sensitive
- 3 attackers, 3 victims
- Goals
  - Precision and speed
  - Resilience to disturbances

Overall evaluation procedure

- **Typing**
  - 3 victims are given the input text
  - Victims type text on their iPhones
- **Recording**
  - A recording camera was used for repeatability
- **Attack**
  - 3 attackers are provided with the videos
  - Attackers have "infinite" time to analyze videos
- **Comparison**
  - Automatic attack vs. human attackers

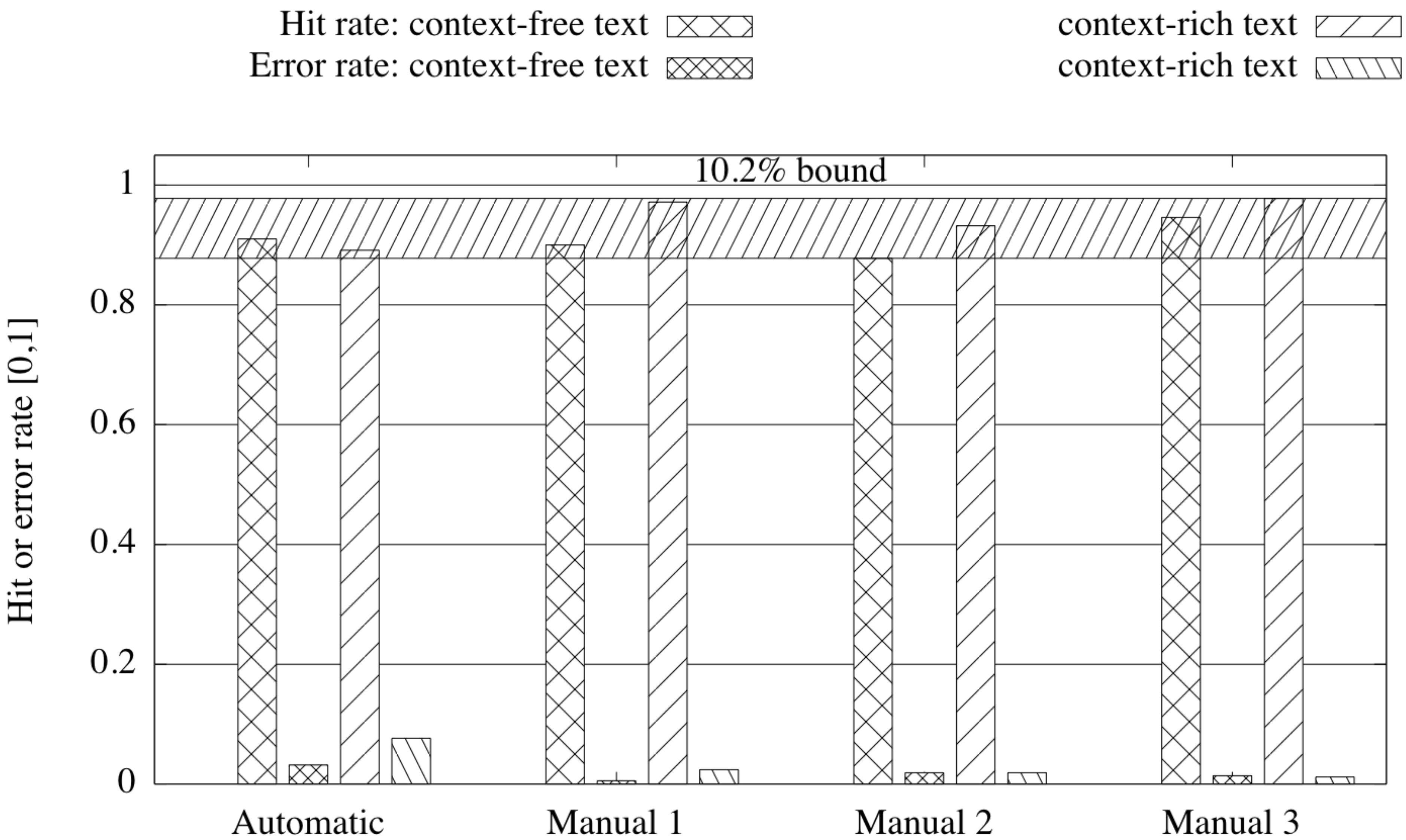
Context-free text

spent chapter foundation identified because first which material notation summarized time spent volume much technical little system reference figured number measurement lorem referring abstract text introductory shown in the we observing request second objective books relationship astute formidable quantile convenient remainder between utilizable tool law resident minutes exemplified the product then temporarily number will per systematic average accumulated south specialty terminal numerous introduce

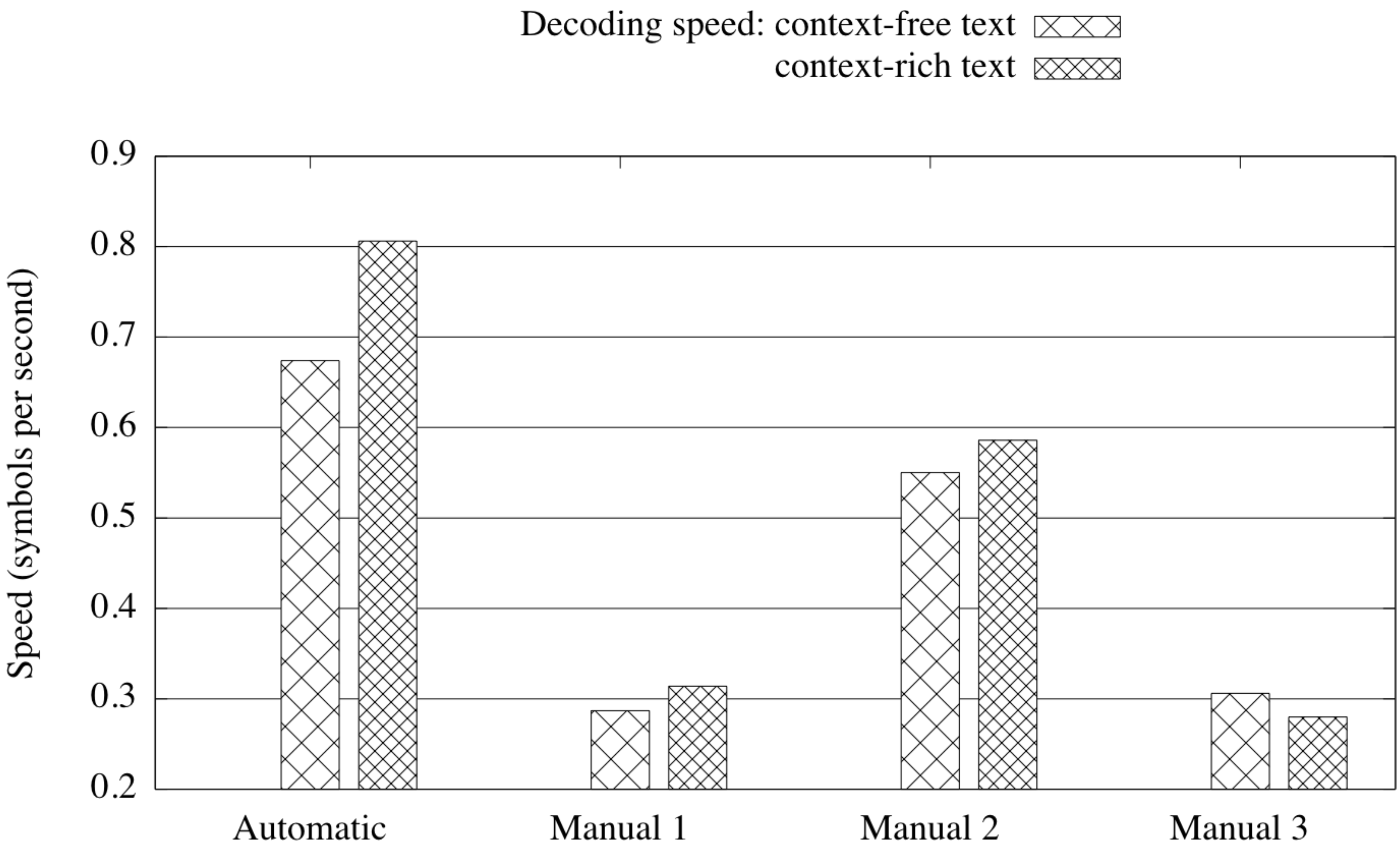
Context-sensitive text

close your eyes and begin to relax take a deep breath and let it out slowly concentrate on your breathing with each breath you become more relaxed imagine a brilliant white light above you focusing on this light as it flows through your body allow yourself to drift off as you fall deeper and deeper into a more relaxed state of mind now as i

Almost as precise as a human



Way faster than a human



Extreme conditions

Aberration	Phase 1	Phase 2-3	
		$h\%$	$\varepsilon\%$
1) Permanent occlusion	difficult	44.44	33.33
2) Shake device	feasible	67.74	8.70
3) Shake camera	feasible	96.00	4.00
4) Shake device + camera	unfeasible	0.00	-

Limitations

- **Non-magnifying keys**
  - Space (on iPhone only)
  - Layout-switching keys
  - **Mitigation**
    - Device-specific heuristics
    - E.g., on iPhone, exploit color-changing spacebar
- **Alternative layouts** (minor limitation)
  - **Mitigation**
    - Detect switch
    - Loop through different templates during detection

Alternative layouts

iSpy: A Happy Coincidence

- [Raguram, CCS 2011]
- Appeared at the same conference
- Completely different approach
  - Classification-based
  - They require training
- Really, the very same accuracy 97~98%

Conclusions

- Touchscreen mobile devices are widespread
- Shoulder surfing is automatable
- Automatic shoulder surfing is precise too
- Counteract these attacks with privacy screens
- But…

Finger tracking

- **Challenge**
  - How to detect tapping?

THANKS!

Federico Maggi
fmaggi@elet.polimi.it

@vp_lab
Dipartimento di Elettronica e Informazione
Politecnico di Milano