Fingerprint Spoof Detector Generalization

IEEE Transactions on Information Forensics and Security (TIFS), 2020

Tarang Chugh
PhD, Michigan State University
Applied Scientist @ Amazon Inc.

Prof. Anil K. Jain
University Distinguished Professor
Michigan State University
Fingerprint Presentation Attack Detection

“presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system”

- ISO standard IEC 30107-1:2016(E)

Silicone  Gelatin  Latex Body Paint  Play Doh  
Gummy Fingers

2-D Printed Spoofs  3-D Printed Spoofs  
Printed Fingerprint Targets

Transplanted skin from sole  Acid Burns  Stitched Fingers  
Altered Fingers

Cadaver Fingers
Fingerprint Spoof Attacks

2D printed paper
3D targets
Conductive ink on paper
Dragon Skin
Gelatin
Gold fingers
Latex body paint
Monster liquid latex
Play doh
Silicone
Transparency
Wood glue
Requirements

- Accurate and Robust

True Detection Rate > 97%
False Detection Rate = 0.2%

Bonafide noisy fingerprint images
Requirements

• Accurate and Robust
• Low-cost and Interoperable

WoodGlue  EcoFlex  Gelatin  Latex  Silgum  Live Fingerprint

Spoof Attacks

Single-finger Readers
SilkID SLK20R  Lumidigm V302

Slap Reader
CrossMatch Guardian 200
Requirements

- Accurate and Robust
- Low-cost and Interoperable
- Efficient

Vivo’s in-display screen fingerprint reader
ZKTeco Access Control Unit
Commodity Smartphone Redmi Note 4 ($150)
Requirements

• Accurate and Robust
• Low-cost and Interoperable
• Efficient
• Interpretable and Generalizable

CNN Fixations

3D t-SNE representation
Proposed Approach: Universal Material Generator
Proposed Approach: Universal Material Generator


\[ AdaIN(x, y) = \sigma(y) \left( \frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y) \]
Proposed Approach: Universal Material Generator


\[ AdaIN(x, y) = \sigma(y) \left( \frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y) \]
Proposed Approach: Universal Material Generator

Proposed Approach: Universal Material Generator

- Style transfer-based wrapper
- Transfer style (texture) characteristics between known PAs

\[ AdaIN(x, y) = \sigma(y) \left( \frac{x - \mu(x)}{\sigma(x)} \right) + \mu(y) \]

### Control Extent of Style Transfer

<table>
<thead>
<tr>
<th>Style A</th>
<th>Synthetic Spoofs</th>
<th>Style B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real Spoof</td>
<td>( \alpha = 0.0 )</td>
<td>Real Spoof</td>
</tr>
<tr>
<td>Latex Body Paint</td>
<td>( \alpha = 0.2 )</td>
<td>Silicone</td>
</tr>
<tr>
<td></td>
<td>( \alpha = 0.4 )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \alpha = 0.6 )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \alpha = 0.8 )</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \alpha = 1.0 )</td>
<td></td>
</tr>
<tr>
<td>Silicone</td>
<td></td>
<td>Latex Body Paint</td>
</tr>
</tbody>
</table>
Proposed Approach: Universal Material Generator

Known Spoof Materials → Universal Material Generator → Generated Patch

Style A + Style B

Style: (0.5A + 0.5B)
Proposed Approach: Universal Material Generator

Known Spoof Materials
+ Style A
+ Style B

Universal Material Generator

Generated Patch
Style: (0.5A + 0.5B)
Proposed Approach: Universal Material Generator

- Known Spoof Materials
- Live Fingerprints
- Synthesized Spoof Fingerprints
- Synthesized Live Fingerprints
- Generated Patch
  Style: $(0.5A + 0.5B)$
Proposed Approach: Universal Material Generator

- Known Spoof Materials
- Live Fingerprints
- Universal Material Generator

Generated Patch
Style: (0.5A + 0.5B)

Synthesized Spoof Fingerprints
Synthesized Live Fingerprints

Spoofs
Lives
Real Spoofs
Real Lives

Concept:
- Content + Style A
- Style B

MICHIGAN STATE UNIVERSITY
Proposed Approach: Universal Material Generator

Known Spoof Materials + Live Fingerprints → Universal Material Generator

Generated Patch Style: (0.5A + 0.5B) → Synthesized Spoof Fingerprints

Spoof Detector → Spoofness Score ∈ [0, 1]

Spoofs → MobileNet-V1 Spoof Detector

Synthetic Spoofs + Real Spoofs → Spoof Detector

Lives → Real Lives

Synthesized Live Fingerprints → Live Fingerprints

Content + Style A → Style B

MICHIGAN STATE UNIVERSITY
Universal Material Generator: Samples

- Style 1
- Synthetic
- Style 2

PA

- Style 1
- Synthetic
- Style 2

Bonafide
Universal Material Generator: Samples

- Style A
  - 2D print
  - 3D targets
  - Conductive ink on paper
  - Dragon skin
  - Gelatin
  - Gold fingers
  - Latex body paint
  - Monster liquid latex
  - Play Doh
  - Silicone
  - Transparency
  - Wood glue

- Style B

Legend:
- Real Live
- Real Spoofs (Known)
- Synthetic Live
- Synthetic Spoofs
- Unknown Spoof (gelatin)
Fingerprint Spoof Generalization: Results

**MSU FPAD-v2 Dataset:** 4,912 spoofs (12 materials), 5,743 live images

<table>
<thead>
<tr>
<th></th>
<th>True Detection Rate @ False Detection Rate = 0.2%</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Base CNN</td>
<td>Base CNN + UMG wrapper</td>
<td></td>
</tr>
<tr>
<td>Slim-Res CNN [2]</td>
<td>73.1 ± 15.7</td>
<td>90.6 ± 10.2</td>
<td></td>
</tr>
<tr>
<td>Fingerprint Spoof Buster [3]</td>
<td>75.2 ± 16.6</td>
<td></td>
<td>91.8 ± 10.3</td>
</tr>
</tbody>
</table>

**LivDet-2017 Dataset:** 9,665 spoofs (6 materials), 8,091 live images

<table>
<thead>
<tr>
<th></th>
<th>True Detection Rate @ False Detection Rate = 1.0%</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Base CNN</td>
<td>Base CNN + UMG wrapper</td>
<td></td>
</tr>
<tr>
<td>Slim-Res CNN [2]</td>
<td>72.6 ± 15.4</td>
<td>78.3 ± 11.9</td>
<td></td>
</tr>
<tr>
<td>Fingerprint Spoof Buster [3]</td>
<td>73.3 ± 15.5</td>
<td></td>
<td>80.7 ± 10.0</td>
</tr>
</tbody>
</table>

Fabricating Unknown Spoofs

Spoof Mixture

(a) Real spoof A (Silicone)
(b) Real spoof B (Latex Body Paint)
(c) Real spoof mixture (spoof A + spoof B)
(d) Synthesized Spoof (spoof A + spoof B)

Performance improved from TDR = 83.33% to 95.83% @ FDR = 0.2%
Summary

• Proposed a style-transfer based wrapper to improve the generalization performance

• Achieved state-of-the-art performance on publicly available LivDet-2017 and MSU FPAD-v2 datasets

• No affect on spoof detection time, < 100ms for Spoof Buster and Slim-Res CNN

• Requires ~2 hours for training UMG wrapper and 1 hour to generate 100,000 patches on Nvidia GTX 1080Ti GPU
Thank You