Automated Cross-Platform Inconsistency Detection for Mobile Apps

Mattia Fazzini

Alessandro Orso

Georgia Tech

Georgia Tech
Mobile Applications
Mobile Applications
Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.
Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.

“Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.”
"Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience."

News

"Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience."

Jul 2016
Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.
Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.
Motivating Example

LG G3

LG Optimus L70
Motivating Example

LG G3

LG Optimus L70
Motivating Example

LG G3

LG Optimus L70
Motivating Example

LG G3

LG Optimus L70
Motivating Example

LG G3

LG Optimus L70
DIFFDROID Overview

Input Generation

Reference Device

App Under Test

Test Case Encoding

Trace

Test Case

Test Devices

CPI Analysis

Test Case Execution

Reference UI Model

Test UI Models

CPI Report

Test Devices
DIFFDROID Overview

Reference Device → Input Generation

App Under Test → Test Case

CPI Analysis → CPI Report

Test Case Encoding

Test Case Execution → Test Devices

Test Case
DIFFDROID Overview

Reference Device

App Under Test

Input Generation

Trace

Test Case Encoding

Test Case

Test Devices

CPI Analysis

Reference UI Model

Test Case Execution

Test UI Models

CPI Report

App
DIFFDROID Overview

Reference Device

App Under Test

Input Generation

Test Case Encoding

Test Case Execution

CPI Analysis

CPI Report

Test Devices

Reference UI Model

Test UI Models

Trace

Test Case

App

UI Models
Input Generation
Input Generation

Reference Device

First, the Reference Device is used to gather inputs, such as key press data, system inputs, and touch interactions. These inputs are then used to build the UI Hierarchy, which represents the layout and structure of the user interface.

The Window Model is used to simulate the display of the interface on different devices, ensuring compatibility and usability across various platforms.

The Screenshots provide visual representations of the interface at different stages of development, allowing developers to assess the design and user experience.

Overall, this process ensures that the generated inputs accurately reflect the intended user interactions, leading to a more intuitive and user-friendly interface.
Input Generation
Input Generation

Reference Device

Inputs

Key
System
Touch

Window Model

UI Hierarchy

Screenshot
Input Generation

Reference Device

Inputs

Key
System
Touch

UI Hierarchy

Screenshot

Trace
Test Case Encoding

Trace

Test Case

Reference
UI Model
Test Case Encoding

Trace → Input → Touch → Test Case

Reference UI Model
Test Case Encoding

Trace → Input → Touch → Test Case

Reference UI Model
Test Case Encoding
Test Case Encoding

Trace → Window Model → UI Hierarchy → Screenshot → Test Case

New

Reference UI Model
Test Case Encoding

Window Model

UI Hierarchy

Screenshot

Trace

Test Case

Reference

UI Model

New
Test Case Execution

Test Case

Test Devices

Execution

Test UI Models
Test Case Execution

Test Case

Test Devices

Execution

Test UI Models
CPI Analysis

Reference Window Model

Test Window Model

Visual Analysis

Structural Analysis

Node Matching

CPI Report

Reference UI Hierarchy

Test UI Hierarchy

Reference Screenshot

Test Screenshot

UI Hierarchy

Screenshot
Structural Analysis

Node Similarity

Reference UI Hierarchy

Test UI Hierarchy

Resource ID

XPath

Properties

- checkable
- focusable
- clickable
- long-clickable
- scrollable
- checked
- focused
- selected
- enabled
- text

Node Matching
Structural Analysis

Node Similarity

Resource ID

XPath

Properties

- checkable: checked
- focusable: focused
- clickable: selected
- long-clickable: enabled
- scrollable: text

Node Matching
Structural Analysis

Node Similarity

Reference UI Hierarchy

Test UI Hierarchy

Resource ID

XPath

Properties
- checkable
- focusable
- clickable
- long-clickable
- scrollable
- checked
- focused
- selected
- enabled
- text

Node Matching
Structural Analysis

Node Similarity

Reference
UI Hierarchy

Test
UI Hierarchy

Resource ID

XPath

Properties

checkable  checked
focusable  focused
clickable  selected
long-clickable  enabled
scrollable  text

Node Matching

Variable

Structural Inconsistency

CPI Report
Visual Analysis

Node Matching

Reference Screenshot | Test Screenshot

Reference Node Image | Test Node Image

CW-SSIM | EMD | RRC | OCR

C4.5 Decision Tree Classifier

Complex-Wavelet Structural Similarity Index (CW-SSIM)

Earth Mover Distance of Color Histogram (EMD)

Relative Ratio Change (RRC)

Optical Character Recognition (OCR)
Visual Analysis

Node Matching

Reference Screenshot

Test Screenshot

Reference Node Image

Test Node Image

CW-SSIM  EMD  RRC  OCR

C4.5 Decision Tree Classifier

Complex-Wavelet Structural Similarity Index (CW-SSIM)

Earth Mover Distance of Color Histogram (EMD)

Relative Ratio Change (RRC)

Optical Character Recognition (OCR)
Visual Analysis

Node Matching

Reference Screenshot

Test Screenshot

Reference Node Image

Test Node Image

Earth Mover Distance of Color Histogram (EMD)

Relative Ratio Change (RRC)

Optical Character Recognition (OCR)

Complex-Wavelet Structural Similarity Index (CW-SSIM)

C4.5 Decision Tree Classifier
Visual Analysis

Node Matching

Reference Screenshot

Test Screenshot

Complex-Wavelet Structural Similarity Index (CW-SSIM)
Earth Mover Distance of Color Histogram (EMD)
Relative Ratio Change (RRC)
Optical Character Recognition (OCR)

C4.5 Decision Tree Classifier
Visual Analysis

Reference Screenshot

Test Screenshot

Node Matching

Reference Node Image

Test Node Image

CW-SSIM  EMD  RRC  OCR

C4.5 Decision Tree Classifier

Visual Inconsistency

Complex-Wavelet Structural Similarity Index (CW-SSIM)

Earth Mover Distance of Color Histogram (EMD)

Relative Ratio Change (RRC)

Optical Character Recognition (OCR)

CPI Report

Visual Inconsistency
Implementation

Input Generation
- Monkey

Test Case Encoding
- Espresso

Test Case Execution
- AWS Device Farm

CPI Analysis
- WEKA
Implementation

Input Generation

Monkey

Test Case Encoding

Espresso

Test Case Execution

AWS Device Farm

CPI Analysis

WEKA
Implementation

Input Generation
- Monkey

Test Case Encoding
- Espresso

Test Case Execution
- AWS Device Farm

CPI Analysis
- WEKA
Implementation

Input Generation
- Monkey

Test Case Encoding
- Espresso

Test Case Execution
- AWS Device Farm

CPI Analysis
- WEKA
Implementation

- Input Generation
  - Monkey
- Test Case Encoding
  - Espresso
- Test Case Execution
  - AWS Device Farm
- CPI Analysis
  - WEKA
Empirical Evaluation

Research Questions:

**RQ1:** Can DIFFDROID detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**RQ2:** What is the cost of running DIFFDROID?

**RQ3:** Are there similarities among devices exhibiting CPIs?
Empirical Evaluation

Research Questions:

RQ1: Can DIFFDROID detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DIFFDROID?

RQ3: Are there similarities among devices exhibiting CPIs?
Empirical Evaluation

Research Questions:

**RQ1:** Can DIFFDROID detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**RQ2:** What is the cost of running DIFFDROID?

**RQ3:** Are there similarities among devices exhibiting CPIs?
Empirical Evaluation

Research Questions:

RQ1: Can DIFFDROID detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DIFFDROID?

RQ3: Are there similarities among devices exhibiting CPIs?
Empirical Evaluation

Research Questions:

**RQ1:** Can DIFFDROID detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**RQ2:** What is the cost of running DIFFDROID?

**RQ3:** Are there similarities among devices exhibiting CPIs?
# Benchmarks and Setup

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Category</th>
<th>Version</th>
<th>LOC (#K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>BuildmLearn</td>
<td>Education</td>
<td>2.5.0</td>
<td>23.6</td>
</tr>
<tr>
<td>A2</td>
<td>Daily Dozen</td>
<td>Health</td>
<td>10.3</td>
<td>6.3</td>
</tr>
<tr>
<td>A3</td>
<td>Kitchen Timer</td>
<td>Tools</td>
<td>1.1.6</td>
<td>4.3</td>
</tr>
<tr>
<td>A4</td>
<td>Outlay</td>
<td>Finance</td>
<td>1.1.3</td>
<td>8</td>
</tr>
<tr>
<td>A5</td>
<td>Translation Studio</td>
<td>Books</td>
<td>9.0</td>
<td>51.2</td>
</tr>
</tbody>
</table>

## Reference Device
LG G3, Android 22

## Test Devices

<table>
<thead>
<tr>
<th>Resolution</th>
<th>OS Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>720x1280</td>
<td>19</td>
</tr>
<tr>
<td>768x1280</td>
<td>21</td>
</tr>
<tr>
<td>1080x1920</td>
<td>22</td>
</tr>
<tr>
<td>1440x2560</td>
<td>23</td>
</tr>
<tr>
<td>480x800</td>
<td>24</td>
</tr>
<tr>
<td>540x960</td>
<td>25</td>
</tr>
<tr>
<td>480x854</td>
<td></td>
</tr>
</tbody>
</table>
# Benchmarks and Setup

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Category</th>
<th>Version</th>
<th>LOC (#K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>BuildmLearn</td>
<td>Education</td>
<td>2.5.0</td>
<td>23.6</td>
</tr>
<tr>
<td>A2</td>
<td>Daily Dozen</td>
<td>Health</td>
<td>10.3</td>
<td>6.3</td>
</tr>
<tr>
<td>A3</td>
<td>Kitchen Timer</td>
<td>Tools</td>
<td>1.1.6</td>
<td>4.3</td>
</tr>
<tr>
<td>A4</td>
<td>Outlay</td>
<td>Finance</td>
<td>1.1.3</td>
<td>8</td>
</tr>
<tr>
<td>A5</td>
<td>Translation Studio</td>
<td>Books</td>
<td>9.0</td>
<td>51.2</td>
</tr>
</tbody>
</table>

## Reference Device

LG G3, Android 22

## Test Devices

<table>
<thead>
<tr>
<th>Resolution</th>
<th>OS Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>720x1280</td>
<td>19</td>
</tr>
<tr>
<td>768x1280</td>
<td>21</td>
</tr>
<tr>
<td>1080x1920</td>
<td>22</td>
</tr>
<tr>
<td>1440x2560</td>
<td>23</td>
</tr>
<tr>
<td>480x800</td>
<td>24</td>
</tr>
<tr>
<td>540x960</td>
<td>25</td>
</tr>
<tr>
<td>480x854</td>
<td></td>
</tr>
</tbody>
</table>
## Benchmarks and Setup

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Category</th>
<th>Version</th>
<th>LOC (#K)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>BuildmLearn</td>
<td>Education</td>
<td>2.5.0</td>
<td>23.6</td>
</tr>
<tr>
<td>A2</td>
<td>Daily Dozen</td>
<td>Health</td>
<td>10.3</td>
<td>6.3</td>
</tr>
<tr>
<td>A3</td>
<td>Kitchen Timer</td>
<td>Tools</td>
<td>1.1.6</td>
<td>4.3</td>
</tr>
<tr>
<td>A4</td>
<td>Outlay</td>
<td>Finance</td>
<td>1.1.3</td>
<td>8</td>
</tr>
<tr>
<td>A5</td>
<td>Translation Studio</td>
<td>Books</td>
<td>9.0</td>
<td>51.2</td>
</tr>
</tbody>
</table>

### Reference Device
- LG G3, Android 22

### Test Devices
- **147**

<table>
<thead>
<tr>
<th>Resolution</th>
<th>OS Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>720x1280</td>
<td>19</td>
</tr>
<tr>
<td>768x1280</td>
<td>21</td>
</tr>
<tr>
<td>1080x1920</td>
<td>22</td>
</tr>
<tr>
<td>1440x2560</td>
<td>23</td>
</tr>
<tr>
<td>480x800</td>
<td>24</td>
</tr>
<tr>
<td>540x960</td>
<td>25</td>
</tr>
<tr>
<td>480x854</td>
<td></td>
</tr>
</tbody>
</table>
**RQ1**

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th>ID</th>
<th>Test Device</th>
<th>Window Model</th>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>135</td>
<td>19</td>
<td>0</td>
<td>2</td>
<td>7</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>A2</td>
<td>138</td>
<td>22</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td>A3</td>
<td>129</td>
<td>13</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A4</td>
<td>125</td>
<td>14</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>A5</td>
<td>136</td>
<td>17</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
</tr>
</tbody>
</table>

**Summary:**
- **Total Structural CPI:** 6
- **Total Functional CPI:** 9
- **Total Version CPI:** 7
- **Total Cosmetic CPI:** 74
- **Total False Positive:** 16
**RQ1**

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th>ID</th>
<th>Test Device</th>
<th>Window Model</th>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>135</td>
<td>19</td>
<td>0</td>
<td>2</td>
<td>7</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>A2</td>
<td>138</td>
<td>22</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td>A3</td>
<td>129</td>
<td>13</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A4</td>
<td>125</td>
<td>14</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>A5</td>
<td>136</td>
<td>17</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
</tr>
</tbody>
</table>

Total: 6 | 9 | 7 | 74 | 16
**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th>ID</th>
<th>Test Device</th>
<th>Window Model</th>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>135</td>
<td>19</td>
<td>0</td>
<td>2</td>
<td>7</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td>A2</td>
<td>138</td>
<td>22</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td>A3</td>
<td>129</td>
<td>13</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>A4</td>
<td>125</td>
<td>14</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>A5</td>
<td>136</td>
<td>17</td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
</tr>
</tbody>
</table>

Sum: 6, 9, 7, 74, 16
**RQ1**

**RQ1**: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

### Example

<table>
<thead>
<tr>
<th>Reference Device</th>
<th>Test Device</th>
</tr>
</thead>
</table>

| Daily Dozen       | 0           |
| Servings         | 0 out of 24 |
| Beans            |             |
| Berries          |             |

<table>
<thead>
<tr>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>7</td>
<td>74</td>
<td>16</td>
</tr>
</tbody>
</table>
RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th></th>
<th>Reference Device</th>
<th>Test Device</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structural CPI</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>2</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>7</td>
</tr>
</tbody>
</table>

| Functional CPI |                  |             |
| 2              | 0                | 0           |
| 2              | 3                | 0           |
| 0              | 1                | 0           |
| 2              | 3                | 0           |
| 6              | 9                | 7           |

| Version CPI    |                  |             |
| 7              | 14               | 1           |
| 0              | 22               | 4           |
| 0              | 2                | 1           |
| 0              | 17               | 2           |
| 0              | 19               | 8           |
| 7              | 74               | 16          |

| Cosmetic CPI   |                  |             |
| 14             | 1                |             |
| 22             | 4                |             |
| 2              | 1                |             |
| 17             | 2                |             |
| 19             | 8                |             |
| 74             | 16               |             |
RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th>Example</th>
<th>Reference Device</th>
<th>Test Device</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>+    +    +    +</td>
<td>+    +    +    +</td>
</tr>
<tr>
<td></td>
<td>23   00   00   -</td>
<td>23   00   00   -</td>
</tr>
<tr>
<td>Start</td>
<td></td>
<td>Start</td>
</tr>
<tr>
<td>Stop</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Table

<table>
<thead>
<tr>
<th></th>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>7</td>
<td>14</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>7</td>
<td>74</td>
<td>16</td>
<td></td>
</tr>
</tbody>
</table>
**RQ1**

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

<table>
<thead>
<tr>
<th>Example</th>
<th>Reference Device</th>
<th>Test Device</th>
<th>Structural CPI</th>
<th>Functional CPI</th>
<th>Version CPI</th>
<th>Cosmetic CPI</th>
<th>False Positive</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>2</td>
<td>7</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>22</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>17</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2</td>
<td>3</td>
<td>0</td>
<td>19</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>6</td>
<td>9</td>
<td>7</td>
<td>74</td>
<td>16</td>
</tr>
</tbody>
</table>
**RQ1**

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?
RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?
RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?
**Evaluation Summary**

<table>
<thead>
<tr>
<th><strong>RQ1</strong>: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFFDROID can detect CPIs in mobile applications while reporting a limited number of false positives</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>RQ2</strong>: What is the cost of running DiffDroid?</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFFDROID can run overnight for the cases considered</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>RQ3</strong>: Are there similarities among devices exhibiting CPIs?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Devices that are more problematic have low values for resolution and density. However, considering testing devices solely based on resolution and density would have not allowed us to identify all the inconsistencies reported.</td>
</tr>
</tbody>
</table>
## Evaluation Summary

<table>
<thead>
<tr>
<th>RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DIFFDROID</strong> can detect CPIs in mobile applications while reporting a limited number of false positives</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RQ2: What is the cost of running DiffDroid?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DIFFDROID</strong> can run overnight for the cases considered</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RQ3: Are there similarities among devices exhibiting CPIs?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Devices that are more problematic have low values for resolution and density. However, considering testing devices solely based on resolution and density would have not allowed us to identify all the inconsistencies reported.</td>
</tr>
</tbody>
</table>
**Evaluation Summary**

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**DIFFDROID** can detect CPIs in mobile applications while reporting a limited number of false positives.

**RQ2:** What is the cost of running DiffDroid?

**DIFFDROID** can run overnight for the cases considered.

**RQ3:** Are there similarities among devices exhibiting CPIs?

Devices that are more problematic have low values for resolution and density. However, considering testing devices solely based on resolution and density would not have allowed us to identify all the inconsistencies reported.
Evaluation Summary

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**DIFFDROID** can detect CPIs in mobile applications while reporting a limited number of false positives

**RQ2:** What is the cost of running DiffDroid?

**DIFFDROID** can run overnight for the cases considered

**RQ3:** Are there similarities among devices exhibiting CPIs?

Devices that are more problematic have low values for resolution and density. However, considering testing devices solely based on resolution and density would have not allowed us to identify all the inconsistencies reported.
Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Summary

Fragmentation

Device

System

Gingerbread
Ice Cream
Jelly Bean
KitKat
Lollipop
Marshmallow
Nougat

News

Jul 2016

“Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall SailfishOS for Android user experience.”

DiffDroid Overview

Input Generation

Reference Device

App Under Test

Test Case Encoding

Trace

Test Case Execution

CPI Analysis

Reference UI Model

Test UI Models

Test Devices

Empirical Evaluation

Research Questions:

RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DiffDroid?

RQ3: Are there similarities among devices exhibiting CPIs?

Future Work

Additional user study

API based differential analysis

Multi-class classifier approach

CPI repair technique
Summary

Fragmentation

Device

System

Gingerbread
Ice Cream
Jelly Bean
KitKat
Lollipop
Marshmallow
Nougat

News

"Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.”

Empirical Evaluation

Research Questions:

RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DiffDroid?

RQ3: Are there similarities among devices exhibiting CPIs?

Future Work

Additional user study

API based differential analysis

Multi-class classifier approach

CPIs repair technique
Summary

Fragmentation

System
- Gingerbread
- Ice Cream
- Jelly Bean
- KitKat
- Lollipop
- Marshmallow
- Nougat

News
July 2016

"Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience."

DiffDroid Overview

Input Generation → Trace → CPI Analysis → Test Case Encoding

Empirical Evaluation

Research Questions:

RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DiffDroid?

RQ3: Are there similarities among devices exhibiting CPIs?

Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique
Summary

Fragmentation

Device

System
- Gingerbread
- Ice Cream
- Jelly Bean
- KitKat
- Lollipop
- Marshmallow
- Nougat

News

Screen

“Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce1 for Android user experience.”

DiffDroid Overview

Input Generation

Trace

Test Case Encoding

Reference UI Model

Test Case Execution

Test Devices

Empirical Evaluation

Research Questions:

RQ1: Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

RQ2: What is the cost of running DiffDroid?

RQ3: Are there similarities among devices exhibiting CPIs?

Future Work

Additional user study

API based differential analysis

Multi-class classifier approach

CPis repair technique
Summary

Fragmentation

Device

System

Gingerbread
Ice Cream
Jelly Bean
KitKat
Lollipop
Marshmallow
Nougat

News

Screen

"Due to the wide array of available Android devices, we are targeting our support to a select number of Android devices to continue improving our overall Salesforce user experience."

Empirical Evaluation

Research Questions:

**RQ1:** Can DiffDroid detect cross-platform inconsistencies in mobile applications while reporting a limited number of false positives?

**RQ2:** What is the cost of running DiffDroid?

**RQ3:** Are there similarities among devices exhibiting CPIs?

Future Work

- Additional user study
- API based differential analysis
- Multi-class classifier approach
- CPIs repair technique