Automated Repair of Layout Cross Browser Issues Using Search-Based Techniques

Sonal Mahajan*, Abdulmajeed Alameer*, Phil McMinn+, William G. J. Halfond*

*University of Southern California, USA
+University of Sheffield, UK

Work supported by NSF Grant CCF-1528163.
Background Information
Background Information

Cross Browser Issues (XBIs) — Inconsistencies in appearance or behavior across browsers.

End-users get a consistent cross-browser user experience

Developers maintain consistent cross-browser user experience
Cross Browser Issues (XBIs)

- 23,000 posts on stackoverflow
- Appearance XBIs are prevalent — 90% [Choudhary et. al. ’13]

**Layout XBIs** — **inconsistent layout of HTML elements in a web page across different browsers**

Layout XBIs appear in 57% of the websites [Choudhary et. al. ‘13]

- Reason for Occurrence
  – Difference in browser implementations of HTML and CSS
Bitcoin is an innovative payment network and a new kind of money.
Welcome to Henry County, Ohio!

Henry County, Ohio is located in the southeast region of Ohio. It has a rich history dating back to the 18th century. The county was established on April 1, 1804, and its name is derived from the Creek Indian word for "river bend." The county seat is Napoleon, and it is part of the Toledo metropolitan area.

The county has a mix of agricultural land, forests, and urban areas. It is home to a number of small towns and villages, including Napoleon, Tinora, and Elida.

The economy of Henry County is based on agriculture, manufacturing, and service industries. Key industries include food processing, automotive parts, and electronics.

The climate of Henry County is characterized by hot summers and cold winters. The area receives about 37 inches of precipitation per year.

For more information, visit the official Henry County website: [Henry County, Ohio](https://www.henrycountyohio.com)

---

<table>
<thead>
<tr>
<th>Elected Officials</th>
<th>Departments, Boards &amp; Commissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>History</td>
<td>Online Auction</td>
</tr>
<tr>
<td>Employment Resources</td>
<td></td>
</tr>
</tbody>
</table>

---

Like Henry County on Facebook!

---

Like Henry County on Facebook!

---

 USC Viterbi  
School of Engineering
Challenges in Repairing XBIs

1. Detection is expensive
   – Large number of browsers and platforms

2. Localization is difficult
   – Complex layouts and styles

3. No standardized ways to repair
   – Resolve XBIs on a case by case basis

4. Repair must not introduce new XBIs
   – Precisely modify problematic UI elements
Challenges in Repairing XBIs

1. Detection is expensive
   – Large number of browsers and platforms

2. Localization is difficult
   – Complex layouts and styles

3. No standardized ways to repair
   – Resolve XBIs on a case by case basis

4. Repair must not introduce new XBIs
   – Precisely modify problematic UI elements
Challenges in Repairing XBIs

1. Detection is expensive
   – Large number of browsers and platforms

2. Localization is difficult
   – Complex layouts and styles

3. No standardized ways to repair
   – Resolve XBIs on a case by case basis

4. Repair must not introduce new XBIs
   – Precisely modify problematic UI elements
Challenges in Repairing XBIs

1. Detection is expensive
   - Large number of browsers and platforms

2. Localization is difficult
   - Complex layouts and styles

3. No standardized ways to repair
   - Resolve XBIs on a case by case basis

4. Repair must not introduce new XBIs
   - Precisely modify problematic UI elements

Goal — Automatically find fixes that can repair the layout XBIs detected in a web page.
Two Key Insights

1. Guided search-based techniques can be used to generate repairs.

Large repair search space

Competing constraints
Two Key Insights

2. Impact of layout XBIs can be quantified by a fitness function leveraging their visual manifestation.

Fitness function computes numeric closeness of candidate solutions to the required solution.

Number of XBIs detected (e.g., using X-PERT)

Layout similarity (size and position of the bounding boxes of HTML elements)
Our Approach

1. Initial XBI Detection
2. Extract Root Causes
3. Search for Candidate Fixes
4. Search for the Best Combination of Candidate Fixes
5. Check Termination Criteria

Page Under Test (PUT)
Ref. Browser
Test Browser
Repaired Page (PUT')
S1. Initial XBI Detection

Obtains set of XBIs when PUT is rendered in reference and test browsers

Descriptor of correct layout position

XPaths of elements showing layout XBI

<top-alignment, <html/body/div[3]/div/div, /html/body/div[3]/div>>
S2. Extract Root Causes

Extracts the root causes relevant to each XBI

\[
<\text{element}, \text{property}, \text{value} >
\]

\[
<\text{label}, <\text{e}_1, \text{e}_2>>
\]

\[
\text{label}_1 \rightarrow \text{p}_1, \ldots, \text{p}_n
\]

\[
\text{label}_n \rightarrow \text{p}_1, \ldots, \text{p}_n
\]
S2. Extract Root Causes

Extracts the root causes relevant to each XBI

\(<\text{element}, \text{property}, \text{value}>\)
S3. Search for Candidate Fixes

Produces individual candidate fixes for each root cause

\[ <\text{element}, \text{property}, \text{value}, \text{value}'> \]

minimize layout deviation of e in reference and test browsers
S3. Search for Candidate Fixes

Produces individual candidate fixes for each root cause

\[ \langle e_{\text{lement}}, p_{\text{roperty}}, v_{\text{alue}} \rangle \rightarrow \text{AVM} \rightarrow \langle e_{\text{lement}}, p_{\text{roperty}}, v_{\text{alue}}, v'_{\text{alue}} \rangle \]

Root cause

Alternating Variable Method (AVM) Search

Candidate fix
Alternating Variable Method

[Korel 1990], [Kempka et. al. 2015]

Exploratory Moves
- Probe neighboring values to establish direction of fitness score improvement
- Add small delta values [-1, 1] and observe impact on fitness

Pattern Moves
- Accelerate fitness score improvements in the established direction
- Add delta values in exponentially increasing step sizes
Fitness Function

Quantify relative layout deviation of PUT rendered in reference and test browsers for every candidate v'.

1. $\Delta pos$: Difference in location

2. $\Delta size$: Difference in size

3. $\Delta npos$: Difference in location of neighbors

Fitness score = $(w1 \times \Delta pos) + (w2 \times \Delta size) + (w3 \times \Delta npos)$
1. $\Delta pos$: Difference in Location

Reference browser

Test browser
1. $\Delta pos$: Difference in Location

Reference browser

Test browser
1. \( \Delta \text{pos}: \) Difference in Location

\[
D_{TL} \leftarrow \sqrt{(x_1^t - x_1^r)^2 + (y_1^t - y_1^r)^2}
\]

\[
D_{BR} \leftarrow \sqrt{(x_2^t - x_2^r)^2 + (y_2^t - y_2^r)^2}
\]

\( \Delta \text{pos} = D_{TL} + D_{BR} \)
1. $\Delta \text{pos}$: Difference in Location

$\Delta \text{pos} = D_{\text{TL}} + D_{\text{BR}}$

$D_{\text{TL}} \leftarrow \sqrt{(x_1^t - x_1^r)^2 + (y_1^t - y_1^r)^2}$

$D_{\text{BR}} \leftarrow \sqrt{(x_2^t - x_2^r)^2 + (y_2^t - y_2^r)^2}$

$\Delta \text{pos}$ decreases as the boxes move closer
2. Δsize: Difference in Size

element

element

Reference browser

Test browser
2. \( \Delta \text{size} \): Difference in Size

\[ \Delta \text{size} = |W_R - W_T| + |H_R - H_T| \]
2. $\Delta$size: Difference in Size

$\Delta$size decreases as the boxes become similar in size

$$\Delta\text{size} = |W_R - W_T| + |H_R - H_T|$$
3. $\Delta npos$: Difference in Location of Neighbors

**Neighbors** — elements that are at a distance of $N$ hops from $e$ in the HTML DOM tree of the PUT

- $\Delta npos$: Difference in Location of Neighbors
3. $\Delta npos$: Difference in Location of Neighbors

**Neighbors** — elements that are at a distance of $N$ hops from $e$ in the HTML DOM tree of the PUT
3. $\Delta n_{\text{pos}}$: Difference in Location of Neighbors

Neighbors — elements that are at a distance of $N$ hops from $e$ in the HTML DOM tree of the PUT.
3. Δ*npos*: Difference in Location of Neighbors

```
<table>
<thead>
<tr>
<th>Reference browser</th>
<th>Test browser</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>e</strong>lement</td>
<td><strong>e</strong>lement</td>
</tr>
<tr>
<td><strong>n</strong>eighbor</td>
<td><strong>n</strong>eighbor</td>
</tr>
</tbody>
</table>
```
3. $\Delta n_{\text{pos}}$: Difference in Location of Neighbors

$D_{TL} \leftarrow \sqrt{(x_1^t - x_1^r)^2 + (y_1^t - y_1^r)^2}$

$D_{BR} \leftarrow \sqrt{(x_2^t - x_2^r)^2 + (y_2^t - y_2^r)^2}$

$\Delta n_{\text{pos}} = D_{TL} + D_{BR}$
3. $\Delta n_{\text{pos}}$: Difference in Location of Neighbors

$D_{TL} \leftarrow \sqrt{(x_1^t - x_1^r)^2 + (y_1^t - y_1^r)^2}$

$D_{BR} \leftarrow \sqrt{(x_2^t - x_2^r)^2 + (y_2^t - y_2^r)^2}$

$\Delta n_{\text{pos}}$ decreases as e’s boxes move closer, causing n’s boxes to also move closer.

$\Delta n_{\text{pos}} = D_{TL} + D_{BR}$
Running Example — S3

Fitness score = (w1 * Δpos) + (w2 * Δsize) + (w3 * Δnpos)
= (1 * 40) + (2 * 0) + (0.5 * 40)
= 60

AVM

Root causes

<html/body/div[3]/div/div, margin-top, -20px>
Running Example — S3

Fitness score = \( w_1 \Delta \text{pos} + w_2 \Delta \text{size} + w_3 \Delta \text{npos} \)

\[
= (1 \times -40 \times 10) + (2 \times 0) + (0.5 \times -40 \times 10)
\]

\[
= -60 \times 15
\]

\[
= 60 + 15
\]

\[
= 75
\]

Root causes

Candidate fixes

AVM

\[
<\/html/body/div[3]/div/div, \text{margin-top, -20px}>\]

\[
<\/html/body/div[3]/div/div, \text{margin-top, -20px, 20px}>\]

margin-top = -20px – 15px
Running Example — S3

Fitness score = (w1 * Δpos) + (w2 * Δsize) + (w3 * Δnpos)
= (1 * -40 - 10) + (2 * 0) + (0.5 * -40 - 10)
= -60 - 15

Root causes

Candidate fixes

<html/body/div[3]/div/div, margin-top, -20px>
<html/body/div[3]/div/div, top, 0px>
<html/body/div[3]/div, margin-top, 0px>
<html/body/div[3]/div, top, 0px>
<html/body/div[3]/div/div, margin-top, -20px, 20px>
<html/body/div[3]/div/div, top, 0px, 20px>
S4. Search for the Best Combination of Candidate Fixes

Finds subset of candidate fixes representing the best overall repair

- Combine candidate fixes to fully resolve an XBI
- Interaction of fixes may have duplicating, multiplying, or competing effects

Candidate fixes

OR

IncredibleIndia
S4. Search for the Best Combination of Candidate Fixes

Finds subset of candidate fixes representing the best overall repair

– Combine candidate fixes to fully resolve an XBI
– Interaction of fixes may have duplicating, multiplying, or competing effects
S4. Search for the Best Combination of Candidate Fixes

Finds subset of candidate fixes representing the best overall repair

Set of candidate fixes

Biased Random Search

Subset of candidate fixes giving the best repair
Biased Random Search

• **Selection**
  – A candidate fix is selected with a probability $\frac{imp_{fix}}{imp_{max}}$

  improvement in fitness in S3
  maximum improvement in S3 across all fixes
Biased Random Search

• Selection
  – A candidate fix is selected with a probability $\frac{imp_{fix}}{imp_{max}}$

• Fitness Function
  – Minimizing goal — number of XBIs

• Termination Conditions
  1. All XBIs are fixed
  2. Maximum threshold of repairs to be tried has been reached
  3. A sequence of repairs with no fitness improvement

X-PERT
[Choudhary et. al. ICSE’13]
Determines whether to terminate or proceed to S2 for another iteration.

Approach terminates if:

1. All XBIs resolved
2. New XBIs same as previous iteration XBIs
3. New number of XBIs more than previous iteration
Produce Repaired Page

Generate test browser specific CSS repair patch

Directs layout engine to use this code if browser is Firefox

```css
1. @-moz-document url-prefix() {
2. html > body > div:nth-of-type(3) > div:nth-of-type(1) > div:nth-of-type(1) {
3.   margin-top: 1.7% !important; /* 20px */
4. }
5. }
```

Repair

```html
<html/body/div[3]/div/div, margin-top, -20px, 20px>
```
Produce Repaired Page

Generate test browser specific CSS repair patch

1. @-moz-document url-prefix() {
2.   html > body > div:nth-of-type(3) > div:nth-of-type(1) > div:nth-of-type(1) {
3.     margin-top: 1.7% !important; /* 20px */
4.   }
5. }

 XPath is converted to CSS selector

</html/body/div[3]/div/div, margin-top, -20px, 20px>

Repair

CSS repair patch
Produce Repaired Page

Generate test browser specific CSS repair patch

1. `@-moz-document url-prefix() {`
2. `html > body > div:nth-of-type(3) > div:nth-of-type(1) > div:nth-of-type(1) {`
3. `margin-top: 1.7% !important; /* 20px */`
4. `}`
5. `}`
Empirical Evaluation

• RQ1: How effective is our approach at reducing layout XBIs?

• RQ2: What is the impact on the cross-browser consistency of the page when the suggested repairs are applied?

• RQ3: How long does our approach take to find repairs?
Experimental Protocol

• Tool
  – Approach implemented in XFix

• Methodology
  – For each subject
    ‣ Select reference and test browsers
    ‣ Run XFix 30 times to mitigate non-determinism in search
  – Human study for impact on cross-browser consistency

RQ1 and RQ3

RQ2
# Subjects

<table>
<thead>
<tr>
<th>Name</th>
<th>#HTML</th>
<th>#CSS</th>
<th>Ref</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>BenjaminLees</td>
<td>317</td>
<td>1,525</td>
<td>CH</td>
<td>FF</td>
</tr>
<tr>
<td>Bitcoin</td>
<td>207</td>
<td>1,957</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>Eboss</td>
<td>439</td>
<td>789</td>
<td>IE</td>
<td>FF</td>
</tr>
<tr>
<td>EquilibriumFans</td>
<td>340</td>
<td>868</td>
<td>CH</td>
<td>FF</td>
</tr>
<tr>
<td>GrantaBooks</td>
<td>325</td>
<td>6,545</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>HenryCountyOhio</td>
<td>300</td>
<td>983</td>
<td>IE</td>
<td>FF</td>
</tr>
<tr>
<td>HotwireHotel</td>
<td>1,457</td>
<td>10,618</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>IncredibleIndia</td>
<td>251</td>
<td>2,172</td>
<td>IE</td>
<td>FF</td>
</tr>
<tr>
<td>Leris</td>
<td>195</td>
<td>1,262</td>
<td>FF</td>
<td>CH</td>
</tr>
<tr>
<td>Minix3</td>
<td>118</td>
<td>821</td>
<td>IE</td>
<td>CH</td>
</tr>
<tr>
<td>Newark</td>
<td>598</td>
<td>17,426</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>Ofa</td>
<td>578</td>
<td>5,381</td>
<td>IE</td>
<td>CH</td>
</tr>
<tr>
<td>PMA</td>
<td>497</td>
<td>13,743</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>StephenHunt</td>
<td>497</td>
<td>13,743</td>
<td>FF</td>
<td>IE</td>
</tr>
<tr>
<td>WIT</td>
<td>300</td>
<td>3,249</td>
<td>FF</td>
<td>IE</td>
</tr>
</tbody>
</table>
RQ1. Reduction of XBIs

Reduction (%) in X-PERT reported XBIs before and after repair
RQ1. Reduction of XBIs

Mean = 86%
Median = 93%
RQ1. Reduction of XBIs

**Hypothesis:** XBIs reported due to pixel-level differences, would not be human perceptible.

**Answer:** Confirmed. Mean 99% (median 100%) reduction in human-observable XBIs.
RQ2. Impact on Cross-Browser Consistency

Similarity (consistency) ratings given by participants of the human study
RQ2. Impact on Cross-Browser Consistency

<table>
<thead>
<tr>
<th>Users</th>
<th>Improved</th>
<th>Same</th>
<th>Decreased</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Improved = 78%
- Same = 14%
- Decreased = 8%
RQ2. Impact on Cross-Browser Consistency

High discordance?

- Inherent browser-level differences
- Text intensive and contain specific fonts rendering differently in browsers

Consistency judgement likely influenced by such differences.
RQ3. Time Needed to Run XFix

Total running time of XFix
RQ3. Time Needed to Run XFix

43 sec

Median = 14 min

2 hours

S3: Search for Candidate Fixes

32%

S4: Search for the Best Combination of Candidate Fixes

1%

Other: S1, S2, and S5

67%

S1, S2, and S5

USC Viterbi
School of Engineering
Artifact

[Image -4x-6 to 1028x123]
[Image 56x401 to 106x446]
[Image 56x300 to 106x345]
[Image 667x617 to 796x750]

https://github.com/sonalmahajan/xfix

- Open source tool release
- Evaluation data
  - Subjects
  - Human study documents
Summary

Challenges in Repairing XBIs

1. Detection is expensive
   - Large number of browsers and platforms
2. Localization is difficult
   - Complex layouts and styles
3. No standardized ways to repair
   - Resolve XBIs on a case by case basis
4. Repair must not introduce new XBIs
   - Precisely modify problematic UI elements

RQ1. Reduction of XBIs

Hypothesis: XBIs reported due to pixel-level differences, would not be human perceptible.

Answer: Confirmed. Mean 99% (median 100%) reduction in human-observable XBIs.

RQ2. Impact on Cross-Browser Consistency

Similarity (consistency) ratings given by participants of the human study

- Improved = 78%
- Same = 14%
- Decreased = 8%
Automated Repair of Layout Cross Browser Issues Using Search-Based Techniques

Sonal Mahajan, Abdulmajeed Alameer, Phil McMinn, William G. J. Halfond

spmahaja@usc.edu
alameer@usc.edu
p.mcminn@sheffield.ac.uk
halfond@usc.edu

Work supported by NSF Grant CCF-1528163
RQ4. Similarity of Repair Patches to Real-world Websites’ Code

Analyzed 480 Alexa websites
RQ4. Similarity of Repair Patches to Real-world Websites’ Code

**Prevalent**

Browser specific code present in 80% websites

```html
@-moz-document url-prefix() {
    html > body > div:nth-of-type(3) {
        height: 90% !important;
        margin-top: 1% !important;
    }
}
```

Size = 2

**Comparable**

XFix generates realistic repair patches

![Graph showing the size of browser specific code for FF, IE, and CH with Avg = 9](image)