User feedback

In this document, we describe the procedure used to elicit user feedback on our flexible anchors compared to the standard approach and the resulting survey. We performed the survey on both touch display and classical keyboard + mouse configurations. To represent the standard approach we used Adobe Photoshop’s Magnetic lasso tool in the classical keyboard + mouse configuration. Due to the OS of our touch device, a node-anchor-only version of our software represented the standard.

Training

We explained to the user how to every anchor available in our system displayed bellow.

The user then train by segmenting the boat in the image bellow, when the user felt comfortable with the system she proceeded to perform the tasks described in the following section.
Tasks

Users were asked to perform the following tasks:

1. Segment the flower using the standard approach (node-only) and our new paint anchor:

![Flower Image]

2. Perform two segmentations of the orange, the full orange then the inner part i.e. without the peel using the standard approach (node-only) and our new flexible anchors:

![Orange Image]
3. Segment the pinwheel with their choice of anchors:

Once the users had performed the prescribed tasks, they were asked to fill the following survey to collect their impressions on our system.

**Survey**

Device Type

<table>
<thead>
<tr>
<th>Touch</th>
<th>Mouse+Keyboard</th>
</tr>
</thead>
</table>

How often do you use image processing software?

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Never</td>
<td>Always</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Compared to the standard approach, how did you find the new approach?

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harder</td>
<td>Easier</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Compared to the standard approach, did you arrive at your desired segmentation:

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Slower</td>
<td>Faster</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
How did you find the flexibility of constraint choice?

1 2 3 4 5

Not Useful Useful

Please rank the following anchors from most (1) to least (5) useful?

____________ Node
____________ Paint
____________ Pinch
____________ Probable
____________ Pick

Would you use a paint, pinch, probable, or pick anchor if available in image processing software?

Yes No

Results

In this section, we present the collected data from the above-mentioned survey.

Device Type

7 Touch 9 Mouse+Keyboard

How often do you use image processing software?

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Never</td>
<td>4</td>
<td>5</td>
<td>3</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Always</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Compared to the standard approach, how did you find the new approach?

- Harder: 0, 0, 3, 7, 6
- Easier: 0, 0, 0, 2, 3

Compared to the standard approach, did you arrive at your desired segmentation:

- Slower: 0, 0, 2, 8, 6
- Faster: 0, 0, 2, 8, 6
How did you find the flexibility of constraint choice?

Please rank the following anchors from most (1) to least (5) useful?

Would you use a paint, pinch, probable, or pick anchor if available in image processing software?

Yes  No
Timings in seconds for the 7 touch screen users

<table>
<thead>
<tr>
<th>Flower standard live-wire</th>
<th>Flower paint anchor</th>
<th>Orange standard live-wire outer+inner</th>
<th>Orange ours both</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>20</td>
<td>15+11</td>
<td>26</td>
</tr>
<tr>
<td>67</td>
<td>30</td>
<td>27+13</td>
<td>40</td>
</tr>
<tr>
<td>32</td>
<td>20</td>
<td>11+9</td>
<td>20</td>
</tr>
<tr>
<td>39</td>
<td>15</td>
<td>15+13</td>
<td>28</td>
</tr>
<tr>
<td>45</td>
<td>37</td>
<td>15+10</td>
<td>25</td>
</tr>
<tr>
<td>49</td>
<td>27</td>
<td>14+16</td>
<td>30</td>
</tr>
<tr>
<td>63</td>
<td>39</td>
<td>13+15</td>
<td>28</td>
</tr>
</tbody>
</table>
Supplemental: Flexible Live-Wire: Image Segmentation with Floating Anchors

B. Summa¹, N. Faraj¹, C. Licorish¹, and V. Pascucci²

¹Tulane University, United States
²SCI Institute, University of Utah, United States

Algorithm 1 UpdateAnchorSet
Interactive update and minimum path lookup during anchor set adjustment:
Input: anchor set: $A_i$; set of new nodes for anchor: $\{n\}$
Output: minimum path: $P$

$A_i ← setNodes(\{n\})$ ★ update nodes
$m ← FindMinCostNode(A_i)$ ★ cost = cost($T_i^F$) + cost($T_i^B$)
$P_i^F ← EXTRACTMINPATH(A_{i−1}, n, F)$ ★ If anchor exists
$P_i^B ← EXTRACTMINPATH(A_{i+1}, n, B)$ ★ If anchor exists
$P_i ← P_i^F ∪ P_i^B$

procedure EXTRACTMINPATH($A_i, n, D$)

$P_i^D ← PATHLOOKUP(T_i^D, n)$ ★ tree traversal from $n$ to root
if $D == F$ then
$P_i^D = P_i^D ∪ EXTRACTMINIMUMPATH(A_{i−1}, n, F)$
else
$P_i^D = P_i^D ∪ EXTRACTMINIMUMPATH(A_{i+1}, n, B)$
end if
return $P_i^D$
end procedure

Algorithm 2 CommitAnchorSet
Tree calculations post user adjustment of an anchor set:
Input: anchor set, $A_i$

procedure COMMITFORWARD($A_i$)
$\{n\} ← getNodes(A_i)$
$C_i^F ← getCosts(\{n\}, T_i^{F−1})$
$Τ_i^F ← COMPUTETREE(\{n\}, C_i^F)$
COMMITFORWARD($A_{i+1}$) ★ If anchor exists
end procedure

procedure COMMITBACKWARD($A_i$)
$\{n\} ← getNodes(A_i)$
$C_i^B ← getCosts(\{n\}, T_i^{B+1})$
$Τ_i^B ← COMPUTETREE(\{n\}, C_i^B)$
COMMITBACKWARD($A_{i−1}$) ★ If anchor exists
end procedure

procedure COMPUTETREE($\{n\}, C$)
InitTreeCalculation($\{n\}, C$) ★ initialize costs to fl. anchor
$Τ ← RunOptimization()$ ★ Dijkstra’s, etc.
return $T$
end procedure