

УДК 004.9

Определение степени удобства активных элементов веб-страницы

Бургина А.М., студент

*Россия, 105005, г. Москва, МГТУ им. Н.Э. Баумана,
кафедра «Программное обеспечение ЭВМ и информационные технологии»*

Научный руководитель: Филиппов М.В., к.т.н., доцент

кафедра «Программное обеспечение ЭВМ и информационные технологии»

bauman@bmstu.ru

При использовании планшетных компьютеров и телефонов пользователь чаще всего взаимодействует с WEB-страницей с помощью сенсорного ввода. Пользователь пальцами нажимает на экран, целясь в элементы страницы, а браузер выполняет различные действия в зависимости от места нажатия. При этом часто возникает ситуация, когда пользователь промахивается и попадает по соседним элементам, или в область нажатия может попасть несколько элементов. В этих случаях могут выполняться незапланированные действия, что замедляет и затрудняет работу пользователя. В настоящее время эта проблема имеет решение в некоторых мобильных браузерах. При их использовании в случае нажатия сразу на несколько элементов, область нажатия увеличивается в размере, и пользователю становится проще попасть в нужный. Однако не во всех ситуациях такой подход возможен. Например, на такие подсказки не могут рассчитывать пользователи сенсорных моноблоков, на которых стоят обычные браузеры, не умеющие давать подсказок.

Определим несколько терминов, которые будут использоваться далее.

Активный элемент - это любая область страницы, при касании которой происходит какое-либо действие, предусмотренное автором. Активными элементами могут, например, являться ссылки, кнопки и изображения.

Успешное взаимодействие - это такое касание пользователя, для которого можно однозначно определить активный элемент, с которым происходит взаимодействие.

Удобство активного элемента будет определяться вероятностью успешного взаимодействия.

1. Модель взаимодействия с активным элементом

Для вычисления удобства построим модель, которая будет описывать взаимодействие пользователя с активным элементом.

Активный элемент является прямоугольником шириной W и высотой H пикселей. Центр прямоугольника примем за начало координат. Оси координат будут направлены вверх и вправо. Сверху, слева, справа и снизу относительно исследуемого активного элемента располагаются другие активные элементы. Расстояние до них определим как D_u , D_l , D_r , D_d пикселей соответственно.

Пользователь нажимает на экран и покрывает некоторую круглую область радиуса R пикселей. Пользователь при нажатии целится в центр исследуемого элемента, но может промахнуться. Примем, что отклонение от центра по каждой координате описывается нормальным распределением, причем отклонения являются независимыми случайными величинами. Будем считать, что пользователь почти всегда попадает центром своего нажатия внутрь активного элемента.

Определим параметры нормальных распределений (m_x, σ_x) и (m_y, σ_y) . Так как по выбранной модели пользователь целится в центр активного элемента, то $m_x = m_y = 0$. Среднеквадратическое отклонение определим из того условия, что пользователь центром своего касания всегда попадает в активный элемент. Воспользуемся «правилом трех сигм», которое утверждает, что практически все ($\approx 99\%$) значения нормального распределения попадают в интервал $(m-3\sigma; m+3\sigma)$. В нашем случае σ будет определяться из размеров активного элемента по формулам (1) и (2).

$$\sigma_x = \frac{W}{6} \quad (1)$$

$$\sigma_y = \frac{H}{6} \quad (2)$$

В данной модели успешное взаимодействие будет происходить тогда, когда в круглую область нажатия пользователя будет попадать только активный элемент, и не будут попадать соседние элементы.

Радиус R можно оценить исходя из наиболее распространенного физического разрешения 96 пикселей на дюйм [1]. Примем размер (диаметр) касания 0.9см [2]. Тогда радиус R в пикселях вычисляется по формуле (3).

$$2R = \left(\frac{0.9}{2.54} \right) * 96 \approx 34 \text{ пикселей} \quad (3)$$

Расположение элементов и величины расстояний изображены на рисунке 1. В центре изображен текущий элемент. Окружность показывает пример области касания пользователя. Внешний прямоугольник задает границы соседних элементов.

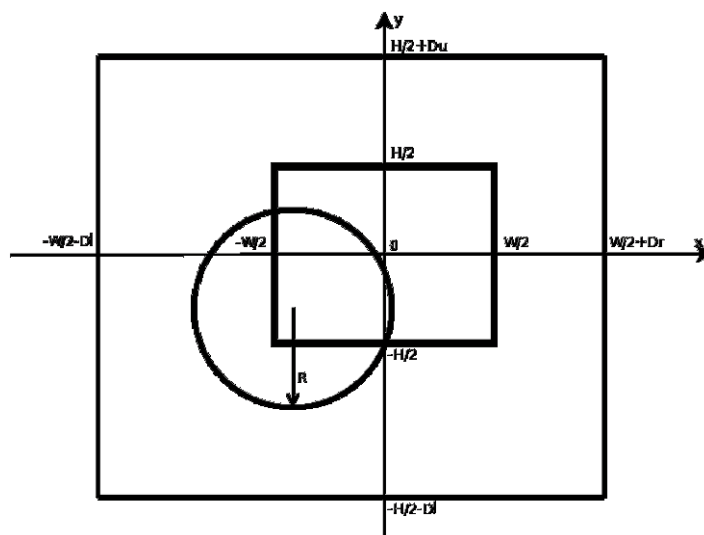


Рис.1. Расположение активного элемента, соседних и касания пользователя

2. Вычисление удобства активного элемента

Обозначим U искомую вероятность успешного взаимодействия. Исходя из описанной модели взаимодействия, U будет вычисляться по формуле (4).

$$U = \iint_S A(x, y) p(x, y) dx dy, \quad (4)$$

где S – область, куда может попасть пользователь при нажатии,

x, y – координаты нажатия пользователя,

$A(x, y)$ – характеристическая функция успеха, которая принимает 1 в случае успешного взаимодействия и 0 в противном случае,

$p(x, y)$ – плотность вероятности нажатия в точке (x, y) .

Область S можно разделить на две составляющие — область с успешным нажатием и область с неуспешным. При этом подынтегральное выражение принимает значение 0 на всей области с неуспешным нажатием. Поэтому из формулы (4) можно убрать функцию $A(x, y)$, изменив область интегрирования (5).

$$U = \iint_S A(x, y) p(x, y) dx dy = \iint_{S_u} p(x, y) dx dy \quad (5)$$

где S_u – область, при нажатии на которую получается успешное взаимодействие.

Определим, из чего состоит область S_u . Успешное взаимодействие будет происходить тогда, когда круглая область нажатия не будет пересекаться с соседними элементами. Для этого необходимо, чтобы расстояние от центра нажатия до соседнего элемента было больше, чем радиус нажатия. Из этого следует, что область успешных нажатий получается путем отступа от соседних элементов «внутри» на расстояние R . Полученный

прямоугольник и является областью успешного взаимодействия S_u . Расположение прямоугольника и его координаты приведены на рисунке 20

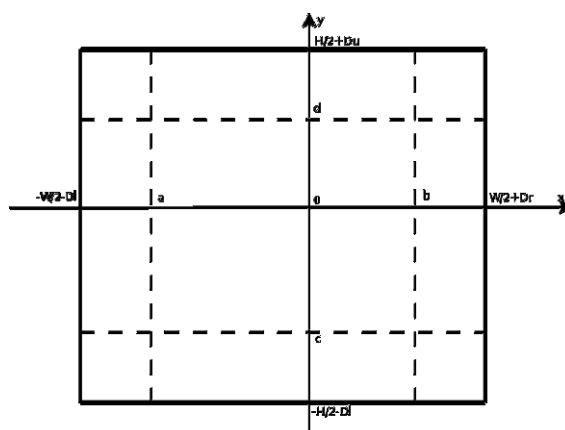


Рис. 2. Границы области успешного взаимодействия.

Заметим, что определение области S_u подобным образом корректно только для случаев, когда расстояние до соседних элементов не превышает радиуса R . Если данное соотношение не выполняется, то граница фигуры S_u будет определяться исходя из радиуса R . В этом случае формы фигуры будет «прямоугольник со скругленными углами», стороны которого будут отступать от выбранного элемента на радиус R . Например, для верхней границы это будет $(H/2+R)$. Пока будем считать, что соседние элементы располагаются достаточно близко, а в дальнейшем покажем, что итоговая полученная формула подходит и для этого случая.

Так как в выбранной модели распределения по координатам приняты независимыми величинами, то $p(x, y) = p(x) \cdot p(y)$.

С учетом определения S_u и преобразования плотности вероятности получаем (6).

$$\iint_{S_u} p(x, y) dx dy = \iint_{\substack{a \leq x \leq b \\ c \leq y \leq d}} p(x)p(y) dx dy = \int_a^b p(x) dx \int_c^d p(y) dy \quad (6)$$

где a, b, c, d задают границы прямоугольника S_u , а именно

$$a = -\frac{W}{2} - D_L + R \quad (7)$$

$$b = \frac{W}{2} + D_R - R \quad (8)$$

$$c = -\frac{H}{2} - D_D + R \quad (9)$$

$$d = \frac{H}{2} + D_U - R \quad (10)$$

Так как распределения нормальные, то можно воспользоваться исследованными свойствами этого распределения [3] и каждый интеграл можно вычислить по формуле (11).

$$\int_a^b p(x) dx = \Phi\left(\frac{b - m_x}{\sigma_x}\right) - \Phi\left(\frac{a - m_x}{\sigma_x}\right) \quad (11)$$

где $\Phi(x)$ – функция Лапласа.

С учетом формулы (11) выражение (6) приобретает вид (12).

$$\begin{aligned} \int_a^b p(x) dx \int_c^d p(y) dy \\ = \left[\Phi\left(\frac{b - m_x}{\sigma_x}\right) - \Phi\left(\frac{a - m_x}{\sigma_x}\right) \right] * \left[\Phi\left(\frac{d - m_y}{\sigma_y}\right) - \Phi\left(\frac{c - m_y}{\sigma_y}\right) \right] \end{aligned} \quad (12)$$

Сведя все преобразования вместе и подставив параметры распределений, полученные в п. 1, получаем итоговую формулу для вычисления удобства элемента (13).

$$U = \left[\Phi\left(\frac{b}{W/6}\right) - \Phi\left(\frac{a}{W/6}\right) \right] * \left[\Phi\left(\frac{d}{H/6}\right) - \Phi\left(\frac{c}{H/6}\right) \right], \quad (13)$$

где a, b, c, d зависят от расположения соседних элементов и вычисляются по формулам (7) – (10), H и W – размеры элемента, Φ — функция Лапласа.

Теперь вернемся к допущению, что соседние элементы располагаются достаточно близко и проверим, как будет себя вести полученная формула, если это не выполняется. Для простоты изложения будем рассматривать верхнюю границу, остальные в силу симметрии ведут себя аналогично. Если верхний сосед располагается слишком далеко, то верхняя граница прямоугольника S_u будет располагаться вне исследуемого элемента. Это приведет к тому, что в интеграл по площади попадут точки вне элемента, в которые пользователь попасть не может исходя из выбранной модели. Однако данные некорректные точки не повлияют на конечный численный результат исходя из выбранного параметра σ_y распределения. Данный параметр как раз выбирался исходя из того, что вероятность попадания внутрь активного элемента близка к 100%. Таким образом, выведенная формула корректно работает для обоих случаев — как для близких соседних элементов, так и для далеких.

3. Итоговый алгоритм

После реализации данный алгоритм был протестирован на реальных сайтах. В качестве порогового значения было выбрано 0.8. То есть если вероятность успешного взаимодействия пользователя с активным элементом меньше 0,8, то элемент отмечается как ошибочный.

Итоговый алгоритм состоит из следующих шагов:

1. Найти все активные элементы.
2. Для каждого активного элемента найти расстояния до соседних активных элементов со всех четырех сторон.
3. Для каждого активного элемента вычислить удобство U по формуле (14).
4. Если $U < 0.8$, то элемент является ошибкой верстки.

Недостатком выбранной модели и алгоритма можно считать допущение, что любые 2 элемента расположены в узлах прямоугольной сетки и у 2х соседних элементов одна из координат одинакова. Не учитываются случаи, когда элементы располагаются со сдвигом. Данное допущение приемлемо для целей тестирования верстки, потому что, как правило, близко расположенные ссылки являются однотипными — элементы меню, управление списком и т. д. Такие однотипные элементы обычно выравниваются в одну строчку или один столбец, что исключает частичные перекрытия.

4. Исследование эффективности работы метода

Для исследования качества работы метода было выбрано 20 веб-страниц. Каждая из них была проанализирована экспертами, а также программой на предмет наличия неудобных активных элементов. Результаты работы представлены в таблице.

Номер страницы	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Ошибок найдено	10	2	2	4	0	16	0	18	3	2	0	0	6	26	0	0	0	73	0	15
Число найденных верно	9	2	2	3	0	16	0	18	3	2	0	0	6	26	0	0	0	62	0	15
Всего ошибок на странице	9	2	2	3	0	16	0	18	3	2	0	0	6	26	0	0	0	62	0	15
Количество пропусков	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Кол-во ложных срабатываний	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0
Полностью верное определение		+	+		+	+	+	+	+	+	+	+	+	+	+	+	+		+	+

Процент страниц с полностью верным определением — 85.

Процент страниц с пропусками — 0.

Процент страниц с ложными срабатываниями — 15.

Пример работы алгоритма представлен на рисунке 3.

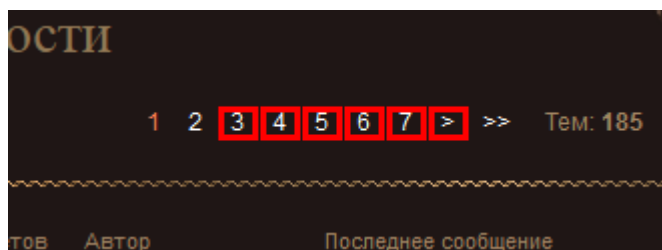


Рис. 3. Пример неудобных элементов, обнаруженных на сайте www.razdor.ru

Алгоритм дает отличные результаты по отсутствию пропусков и незначительное количество ложных срабатываний, которые связаны с описанным допущением, что в алгоритме предполагается, что у двух соседних элементов одна из координат одинакова. В небольшом количестве случаев присутствует сдвиг, но, как и предполагалось, количество таких случаев мало и в основном близко расположенные активные элементы являются выровненными пунктами меню и ссылками на страницы списков.

Стоит отметить, что все большее количество сайтов сейчас делаются адаптируемыми под сенсорное взаимодействие. Во время подготовки данной статьи поменяли свой дизайн и сделали его более «планшетным» сразу два крупных сайта - www.livejournal.ru и habrahabr.ru. И если старый дизайн первого сайта содержал 53 ошибки, связанные со слишком близким расположением элементов, то в новом дизайне их нет ни одной.

Список литературы

1. Pixel Density and Usability. Available at: [http://msdn.microsoft.com/en-us/library/ms699473\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/ms699473(v=vs.85).aspx), accessed 20.05.2014.
2. Ericson J. Improving performance, accuracy, and reliability in touch-screen based applications. Available at: <http://embedded-computing.com/articles/improving-accuracy-reliability-touch-screen-based-applications>, accessed 20.05.2014.
3. Корн Г., Корн Т. Справочник по математике для научных работников и инженеров. Определения, теоремы, формулы / под общей ред. И.Г. Арамановича. 4-е изд.. М.: Наука, 1977. 832 с.