

как следствие, долгая болезненная защита работы с проверкой преподавателя на прочность. При выполнении *расчетно-графических работ* студенты 1-й группы стараются по возможности раньше сдать работу, что обеспечивает им дополнительные баллы студенты 2-й группы стараются сдать работы вовремя чтобы не потерять баллы за просроченные дни; студенты 3-й группы - просто стараются сдать работы

Рейтинговый контроль как средство самоконтроля и активизации учебной деятельности студентов. Так как студенты третьей группы обычно апатичны ко всему происходящему, то рейтинговая система фактически оказывает воздействие на студентов первой и второй групп. Имея желание иметь хорошую рейтинговую отметку (а она впоследствии влияет и на экзаменационную), такие студенты следят за своей успеваемостью по предмету. Не плохую роль в активизации учебной деятельности студентов играет их амбициозность: чем больших результатов хочет добиться студент, тем больше усилий он прилагает. Помогает и соревновательный дух рейтинговой системы: лидеры рейтингового листа активизируются на удержание достигнутых результатов, следующие же за ними, чтобы сменить своих сокурсников на верхних строчках рейтингового листа. Таким образом, рейтинговая система позволяет студенту взглянуть на свою успеваемость посредством сравнения с успеваемостью своих товарищей и разобраться, что он мог сделать лучше, чем сделал.

Таким образом, рейтинговый контроль знаний позволяет студентам более рационально организовать собственный учебный процесс по дисциплине, способствует активизации учебной деятельности обучаемого, развивает навыки самоконтроля в обучении. В своем большинстве студенты положительно относятся к рейтинговому методу контроля знаний

ЛИТЕРАТУРА:

1. Маркевич, К.М., Троян, Т.А. Мониторинг качества обучения на основе комплексной функции контроля знаний студентов //Высшая школа: проблемы и перспективы: VI международная научно-методическая конференция РИВШ. – Минск, 2004, – С 237–239 и др
2. Маркевич, К.М. Рейтинговый контроль при преподавании дисциплин физического профиля. //Вестник Брестского государственного университета – 1999 – № 4 – С 51–58

УДК 004.514.6

Борушко И.Н., Гоманова Е.В.

Научный руководитель: Костюк Д.А., к.т.н.

ПРИМЕНЕНИЕ МОДЕЛИ ПЕРИФЕРИЧЕСКОГО ЗРЕНИЯ В АППАРАТНО-УСКОРЕННОМ ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ

Джеф Раскин в [1] сравнивает с лабиринтом ориентирование в современном программном интерфейсе, когда пользователь не имеет возможности видеть одновременно, хотя бы схематично, изображение всего рабочего пространства. В основном такой подход к организации интерфейса вызван ограниченностью аппаратных ресурсов персонального компьютера, не позволяющих задействовать большие площади для вывода информации. В современных программных продуктах используется ряд решений, призванных избавить пользователя от необходимости сознательного удерживания в памяти схемы «лабиринта»

В данной статье предложены способы увеличения рабочей области, основанные на моделях интерфейса, копирующие разделение поля зрения человека на центральную и периферийную зоны.

Периферическое зрение (ПЗ) называемое часто также боковым или палочковым зрением, играет важную роль в ориентировании человека в окружающей среде. Отправным пунктом для разработанных интерфейсных решений послужила разница в разрешающей способности рецепторов, участвующих в периферическом и в центральном зрении

Уменьшение разрешающей способности по мере приближения к периферии видимого пространства смоделировано нами применительно к ГПИ в виде разделения рабочей области на три части: центральную (2) и боковые (1 и 3). В центральной части, имеющей стандартное разрешение, расположены элементы интерфейса, с которыми пользователь работает непосредственно в данный момент. В областях 1 и 3 разрешение по горизонтали понижается в направлении от центра:

$$\Delta x = \begin{cases} C, x \in (2) \\ C \alpha(x), x \in (1), x \in (3) \end{cases} \quad (1)$$

где Δx – размер зерна (расстояние между центрами отдельных фотоприемников и, соответственно, отдельных точек изображения), C – константа, определяющая единичный размер зерна, соответствующий области 2, $\alpha(x)$ – функция пространственных искажений [2], монотонно возрастающая и имеющая область значений, принадлежащую интервалу $(0; 1]$. Для простоты считаем, что разрешающая способность не изменяется по вертикали.

На практике пользователь наблюдает переменный масштаб изображений, находящихся в периферийных областях, а сами периферийные области, помещаясь на экране целиком, занимают на нем существенно меньше места, чем в виртуальном рабочем пространстве. Это позволяет компактно располагать элементы ГПИ, задействованные в работе лишь косвенно. Тем самым пользователь получает возможность облегченной ориентации в сложной графической среде. Кроме того, события, происходящие на периферии рабочей области, достаточно легко отслеживать, поскольку, как и в случае ПЗ, пониженная разрешающая способность не слишком ухудшает восприятие движения.

Ниже описаны две модели, реализующие переменный масштаб для оконного интерфейса, лежащего в основе всех современных графических систем.

Дискретное изменение масштаба. Мини-окна

В первой из разработанных моделей в качестве опорной точки для выбора масштаба окна принимается абсцисса (горизонтальная координата) его левого верхнего угла. Рабочий стол разделяется на три неравные области. В центральной области, соответствующей прямому зрению, коррекция размеров окна не производится. Благодаря этому пользователь получает возможность работать с приложениями в масштабе 1:1, без нарушения четкости текста и др. В разработанной модели центральная область занимает 2/3 экрана. По бокам расположены области, которым соответствует переменный коэффициент масштабирования окон.

Поскольку под окном понимается прямоугольная область экрана, масштаб изменяется дискретно, т.е. каждая точка, принадлежащая данному окну, отображается в масштабе, соответствующем точке x_0, y_0 (левый верхний угол окна)

$$x_1 = x, \alpha(x_0) [\theta(x_1) - \theta(x, -x_0)] \quad (2)$$

$$y_1 = y, \alpha(x_0) [\theta(y_1) - \theta(y, -y_0)] \quad (3)$$

где α – функция, задающая коэффициент сжатия в зависимости от координаты.

Теоретически возможны два варианта масштабирования: масштабирование в одном измерении, соответствующем оси абсцисс, и двумерное изменение масштаба. В первом случае окна, находящиеся в боковых областях, изменяют свою ширину, сохраняя высоту. Во втором изменяются оба геометрических параметра окна.

В данной модели пользователь сам решает, какие окна размещать в периферийных областях, а какие в центральной. В связи с этим представляется целесообразным применение аналога программы пейджера. Пейджер должен по требованию пользователя сохранять в кольцевом списке схемы расположения окон на рабочей области. Таким образом, расположив в начале работы окна, пользователь имеет возможность сохранить нужные ему схемы расположения, и затем использовать пейджер для быстрого переключения между ними.

Плавное изменение масштаба

В реальном поле зрения человека нельзя выделить три дискретные области с различным разрешением, поскольку концентрация колбочек и палочек на сетчатке изменяется плавно, и зоны центрального зрения и ПЗ постепенно переходят одна в другую. Тем более нельзя говорить о скачкообразном изменении детализации перемещающихся в поле зрения объектов. Можно считать, что масштаб видимого изображения изменяется от центра к периферии, затрагивая не отдельные элементы, а все поле зрения целиком.

Масштаб рабочей области также можно плавно изменять на периферийных областях, сохраняя соотношение 1:1 в центральной области. Математическое выражение для пересчета координат, обобщенное для размерности $k = \overline{1, n}$, выглядит следующим образом:

$$\begin{aligned} P_i^{(k)} &= E_i^{(k)} + F_i^{(k)} + G_i^{(k)}, \\ E_i^{(k)} &= \alpha \left(\frac{P_i^{(k)}}{P_{src}^{(k)}} \right) \cdot \delta \cdot \left[\theta(P_i^{(k)}) - \theta(P_{src}^{(k)} - \delta) \right], \\ F_i^{(k)} &= (P_i^{(k)} - P_{src}^{(k)}) \cdot \left[\theta(P_i^{(k)} - P_{src}^{(k)} - \delta) - \theta(P_i^{(k)} - P_W^{(k)} + \delta - P_{src}^{(k)}) \right], \\ G_i^{(k)} &= \left[P_W^{(k)} - \alpha \left(1 - \frac{P_i^{(k)} - P_W^{(k)} - P_{src}^{(k)} + \delta}{P_{Wsrc}^{(k)} - P_W^{(k)} - P_{src}^{(k)} + \delta} \right) \cdot \delta \right] \cdot \left[\theta(P_i^{(k)} - P_W^{(k)} + \delta - P_{src}^{(k)}) - \theta(P_i^{(k)} - P_{Wsrc}^{(k)}) \right]. \end{aligned} \quad (4)$$

Здесь периферийные области обозначены как E , G , а центральная — F ; θ — функция Хевисайда, α — функция изменения масштаба, δ — ширина области, отведенной под скроллинг. Для исходного изображения $i = 0; W_{src} - 1$, а для результирующего $i = 0; w - 1$ (величины и размерности, относящиеся к исходному изображению, отмечены индексом src). $P_{src}^{(k)}$ — сдвиг левого верхнего угла (соответственно точки локальной координатой 0) результирующего изображения (т.е. окна) относительно аналогичной точки исходного.

В переходе к двумерным координатам $P_i^{(1)} = x_i, P_i^{(2)} = y_i, k = \overline{1, 2}$, рассматривая (4) как $P_i^{(k)} = f(P_i^{(k)}, \delta, P_W^{(k)}, P_{src}^{(k)}, P_{Wsrc}^{(k)})$, получаем:

$$P_i = f \left(P_i = \begin{bmatrix} x_i \\ y_i \end{bmatrix}, \delta, P_W^{(k)} = \begin{bmatrix} W \\ h \end{bmatrix}, P_{src}^{(k)} = \begin{bmatrix} X_{src} \\ Y_{src} \end{bmatrix}, P_{Wsrc}^{(k)} = \begin{bmatrix} W_{src} \\ h_{src} \end{bmatrix} \right) \quad (4')$$

где w_{src}, h_{src} — ширина и высота исходного окна; w, h — ширина и высота результирующего окна.

Выбор функции изменения масштаба α из выражения (2) имеет существенное значение. В частности, использование нелинейной функции позволяет лучше видеть элементы, расположенные ближе к центральной области, и сильнее скрадывает находящиеся на большем удалении. Для разработанных моделей был опробован и линейный вариант, однако впоследствии выбор был сделан в пользу функции вида $\alpha(x) = x^{1/K}$, где $x \in [0; 1]$, K — коэффициент нелинейности, равный отношению ширины области результирующего окна, отведенной под прокрутку, к ширине соответствующей ей сжимаемой области исходного окна. В терминах выражений (2–3) коэффициент нелинейности запишется как

$$\begin{aligned} K &= \left[\frac{\delta}{P_{src}^{(k)} + \delta} \right] \cdot \left[\theta(P_i^{(k)}) - \theta(P_i^{(k)} - P_{src}^{(k)} - \delta) \right] + \\ &+ \left[\frac{\delta}{P_{Wsrc}^{(k)} - P_W^{(k)} - P_{src}^{(k)} + \delta} \right] \cdot \left[\theta(P_i^{(k)} - P_W^{(k)} + \delta - P_{src}^{(k)}) - \theta(P_i^{(k)} - P_{Wsrc}^{(k)}) \right] \end{aligned} \quad (5)$$

Данный подход можно использовать для отображения увеличенной ширины рабочей стола. Однако может оказаться целесообразным его использование на различных

уровнях детализации ГПИ, где нет возможности вместить все изображение целиком и потому необходима прокрутка. Причина в общеизвестном недостатке полос прокрутки с точки зрения эргономики.

Использование нелинейного масштабирования периферийных областей вместо прокрутки позволяет визуально ориентироваться при поиске элемента, находящегося вне центральной (рабочей) части окна.

Интерфейс с аппаратно-ускоренной графикой

Традиционный способ вывода окон экрана в выделенные им области – использование прямого или обратного алгоритма закрашивания. В случае прямого алгоритма каждый последующий слой наносится на предыдущий. Сначала отрисовывается фон, а затем каждое окно поверх предыдущего, как бы нанизываясь на ось Z, направленную перпендикулярно плоскости экрана. В случае обратного алгоритма сначала выводится самое верхнее окно, а затем каждое следующее, используя прямоугольники отсечения – таким образом, чтобы каждый пиксель наносился на экран только один раз. Данные алгоритмы имеют существенный недостаток – в некоторых случаях алгоритм не может точно вычислить порядок расположения окон относительно оси Z, что не позволяет реализовать полноценный трехмерный интерфейс.

При появлении поддержки аппаратных возможностей современного видеоадаптера в отрисовке окон наметились сдвиги. Например, расширение Composite для X Window System по-прежнему использует все тот же закрашивающий алгоритм, но, отрисовка окон делается во внеэкранный, невидимую область видеопамати. Затем оконный менеджер составляет из этих окон изображение для видимой части. Поскольку оконный менеджер имеет в своем распоряжении содержимое всех окон, становится возможным организовать частичную прозрачность и некоторые другие эффекты.

На основе анализа экспериментальных интерфейсных решений, использующих аппаратное ускорение трехмерной графики, разработана реализация предлагаемых моделей, позволяющая использовать основанные на них рабочие среды с уже существующим программным обеспечением без модификации последнего, за счет перехвата изображения и его обработки в реальном масштабе времени графическим процессором стандартной видео-карты. Для реализации моделей ПЗ было выбрано расширение Xgl системы X Window System и кодовая база оконного менеджера Compriz [3]. Выбор в пользу Xgl обоснован его более стабильной работой по сравнению с аналогом и меньшей избирательностью к аппаратному обеспечению. На выбор Compriz повлияла его модульная структура, позволяющая реализовать желаемые возможности в отдельных модулях, обходясь минимальными изменениями остального кода. Так, на основе его расширяемой архитектуры относительно несложно реализовать код, который будет выполнять масштабирование окон по нужному алгоритму, соответствующему разработанному в данной работе моделям.

ЛИТЕРАТУРА:

1. Раскин Дж. Интерфейс: новые направления в проектировании компьютерных систем. СПб.: Символ-Плюс, 2003. 272 стр.
2. Прэтт У. Цифровая обработка изображений. Т. 1. М.: "Мир", 1982. 310 стр.
3. Tamponi E. Communication between Xorg, Xgl, and an OpenGL client, through libGL and the GLX Protocol. 2006. <http://principe.homelinux.net>

УДК 004.514.62

Борушко И.Н., Гоманова Е.В.

Научный руководитель: Костюк Д.А., к.т.н.

АНАЛОГИЯ ПЕРИФЕРИЧЕСКОГО ЗРЕНИЯ В ГРАФИЧЕСКОМ ИНТЕРФЕЙСЕ ПОЛЬЗОВАТЕЛЯ

Периферическое зрение (ПЗ), называемое часто также боковым или палочковым зрением, играет важную роль в ориентировании человека в окружающей среде. Соглас-