

Н. Ю. Рязанова, В. А. Ульихин

**ВОПРОСЫ МАСШТАБИРОВАНИЯ
ИЗОБРАЖЕНИЙ С УЧЕТОМ ИХ СОДЕРЖАНИЯ**

Рассмотрены вопросы выбора алгоритмов выявления сюжета и определения значимости частей отображаемой сцены путем выделения контуров изображаемых объектов. Полученная информация является основой для построения последовательности преобразований, позволяющих в реальном времени при изменении размеров изображения перестраивать его таким образом, чтобы сохранялись пропорции основных объектов сцены.

E-mail: ryaз_nu@mail.ru

Ключевые слова: адаптивное масштабирование изображения, выделенная линия, содержание, основанное на значимости, удаление непрерывных последовательностей пикселей.

Анализ проблемы контекстного масштабирования изображений. Проблема непропорционального масштабирования изображений (например, свободное трансформирование в графическом редакторе Photoshop), когда надо подогнать размеры рисунка под размеры выделенного окна или вписать изображение в заданную рамку, связана с изменением пропорций объектов, составляющих обрабатываемую сцену. Например, при сжатии изображения сцены, содержащей фигуру человека на переднем плане, по ширине приведет к изменению пропорций фигуры, которая при том же росте станет уже. При уменьшении картинка по высоте искажения будут еще больше с точки зрения визуального восприятия: возникнет эффект кривого зеркала — фигура при той же ширине уменьшится в два раза по высоте. При непропорциональном растяжении картинка в высоту или ширину возникают аналогичные искажения формы.

Примером решения данной проблемы может служить алгоритм Seam Carving, разработанный в исследовательской лаборатории Mitsubishi Electric [1], по которому выполняется масштабирование с учетом содержания (от content-aware resizing — *англ.*). Применение алгоритма позволяет изменять соотношение сторон картинка без соответствующего искажения размеров объектов, составляющих изображенную сцену. В основу алгоритма положен принцип учета важности, заметности точек изображения. Для этого рассчитывается вес/удельный вес/энергия точки. Очевидно, что для удаления пикселей следует выбирать самые незаметные, мало значащие пиксели с минимальной энергией.

Известно, что наиболее информативными составляющими изображения являются контуры изображенных объектов. На контурах изображения происходит резкое изменение яркости. Незаметными, малозначимыми с точки зрения восприятия являются точки участков фона, имеющие постоянную или медленно изменяющуюся яркость. Удаление отдельных, никак не связанных между собой пикселей не решает проблему. Необходимо выбрать для удаления (или вставки) такие участки изображения, которые не изменяют форму содержащихся на нем объектов. В алгоритме Seam Carving предлагается удалять (или вставлять) вертикальную или горизонтальную непрерывную последовательность пикселей — seam (шов — *англ.*). “Шов” должен быть шириной в один пиксел, содержать только соседние пиксели изображения, т.е. пиксели, соприкасающиеся сторонами или углами, быть непрерывным и пересекать всю картинку. Такую последовательность назовем связанной последовательностью пикселей.

Для “растягивания” изображения полученные связанные последовательности пикселей дублируются на изображении. Для “сжатия” — связанные последовательности пикселей удаляются (буквально Seam Carving означает “вырезание швов” — *англ.*). В результате значимые детали изображения воспроизводятся без искажений и содержание сцены не изменяется.

Последовательность преобразований. Для нахождения цепочки пикселей с наименьшей суммарной значимостью необходимо выделить на изображении границы объектов — их контуры. Задача выделения контуров состоит в построении дополнительного изображения, содержащего только контуры изображения. Для этого изображение можно преобразовать в черно-белое того же разрешения, в котором яркость и цвет каждого пиксела заменяется соответствующим значением оттенка серого (от 1 до 256).

Как правило, граница предмета на фотографии отображается перепадом яркости между двумя сравнительно однородными областями. Однако перепад яркости может быть связан с текстурой предмета, отбрасываемыми предметами тенями или, наоборот, бликами на поверхностях объектов. Таким образом, выделение контуров чувствительно к шуму. Наличие шума может привести к появлению ложных контуров, не являющихся границами областей с постоянной или медленно изменяющейся яркостью. Удаление шума следует выполнять так, чтобы сохранить важные для последующего семантического описания/распознавания детали изображения. Для шумоподавления или сглаживания используются разные методы, как правило, хорошо работающие с определенными видами шумов. Анализ показывает, что наиболее адекватными практически получаемым шумам являются модели адитивного гауссового шума и импульсного шума [2].

После сглаживания производится выделение контуров, для чего используются детекторы краев и методы связывания точек контура. Сформировав карту краев, можно перейти к поиску связных цепочек пикселей, имеющих минимальные суммы по карте значимости. Затем полученные связные цепочки пикселей используются для контекстного масштабирования.

Способы подавление шумов. Линейное усреднение. Простейшая идея сглаживания состоит в усреднении значений яркостей соседних пикселей. Берется так называемая апертура фильтра — окно (часть изображения), с которым работает фильтр. Апертура постепенно передвигается по изображению слева направо и сверху вниз на один пиксел. В результате на следующем шаге фильтр работает с окном, в котором находятся не только элементы исходного изображения, но и элементы, ранее подвергнутые преобразованию. Чем больше апертура, тем сильнее усреднение. Самый простой вариант фильтрации — для заданного размера апертуры вычислять среднее арифметическое значение всех пикселей апертуры вокруг центрального пикселя, которому затем и присваивается полученное значение. Интересная модификация этого метода была разработана де Хааном, который предложил брать соседние пиксели не подряд, а через один или два пикселя. Утверждается, что при таком подходе подавляется низкочастотный шум, который визуально более заметен, чем высокочастотный [3].

Медианный фильтр. Данный способ основывается на нахождении медианы — среднего элемента, а не среднего арифметического последовательности. Для этого элементы апертуры упорядочиваются. Например, для апертуры 3×3 нужно упорядочить 9 точек. Затем значение пятого элемента упорядоченной последовательности присваивается центральному пикселу окна фильтра.

Сглаживание с помощью гауссиана. Дискретное гауссово ядро сглаживания (апертуру фильтра) можно получить, построив массив размером $(2k + 1) \times (2k + 1)$, значение элемента (i, j) которого:

$$H_{ij} = \frac{1}{2\pi\sigma} e^{-\frac{(i-k-1)^2 + (j-k-1)^2}{2\sigma}},$$

где σ — это среднеквадратическое отклонение гауссиана, параметр, который задает степень размытия. Данное ядро сглаживания образует такое взвешенное среднее, для которого в центре ядра весовые коэффициенты пикселей намного больше, чем на его границах. Маска фильтра такова, что центральный элемент маски имеет наибольшее значение, он соответствует пику распределения Гаусса. Значения остальных элементов уменьшаются по мере удаления от центрального элемента, что происходит в соответствии с распределением Гаусса.

Следующий пример демонстрирует маску фильтра Гаусса размером 5×5 со среднеквадратическим отклонением 0,5 [4]:

0,0000	0,0000	0,0002	0,0000	0,0000
0,0000	0,0113	0,0837	0,0113	0,0000
0,0002	0,0837	0,6187	0,0837	0,0002
0,0000	0,0113	0,0837	0,0113	0,0000
0,0000	0,0000	0,0002	0,0000	0,0000.

Методы выделения контуров. Под контуром изображения понимают пространственно протяженный разрыв, перепад или скачкообразное изменение значений яркости. Для надежного определения точки контура как точки, находящейся на границе перепада яркости, изменение яркости, ассоциированное с данной точкой, должно быть существенно бóльшим, чем изменение яркости в точке фона. Яркость точки оценивается по отношению к соседним с ней точкам, т. е. анализируется локальная область изображения. Поэтому определение того, какое значение является существенным, а какое — нет, состоит в установлении величины порога.

Одним из наиболее соответствующих природе изображений способов обнаружения границ является дифференцирование яркости, которая рассматривается как функция пространственных координат [5].

Дифференциальные методы. Найти контур изображения со значениями яркости $f(x, y)$ можно путем взятия частных производных df/dx и df/dy , которыми оценивают скорость изменения яркости в направлениях x и y соответственно. На практике требуется выделять контуры, направление которых является произвольным. С этой целью используется модуль градиента функции яркости:

$$|\nabla f(x, y)| = \sqrt{\left(\frac{df}{dx}\right)^2 + \left(\frac{df}{dy}\right)^2},$$

который пропорционален максимальной (по направлению) скорости изменения функции яркости в данной точке и не зависит от направления контура. Модуль градиента в отличие от частных производных принимает только неотрицательные значения, поэтому на получающемся изображении точки, соответствующие контурам, имеют уровень яркости белого.

Для цифровых изображений вместо частных производных берутся разностные функции [6]:

$$\frac{df(x, y)}{dx} \approx s_1(n_1, n_2) = f(n_1, n_2) - f(n_1 - 1, n_2),$$
$$\frac{df(x, y)}{dy} \approx s_2(n_1, n_2) = f(n_1, n_2) - f(n_1, n_2 - 1).$$

Для увеличения скорости вычисления модуля градиента можно воспользоваться следующей функцией:

$$g(n_1, n_2) = |s_1(n_1, n_2)| + |s_2(n_1, n_2)|.$$

Операторы Собела и Прюитта. По своей сути эти операторы являются дискретными дифференциальными операторами, которые вычисляют приближенное значение градиента яркости изображения. Для этого операторы используют маски 3×3 , с помощью которых сворачивают исходное изображение для вычисления приближенных значений производных по горизонтали и вертикали. У оператора Собела влияние шума угловых элементов несколько меньше, чем у оператора Прюитта, что существенно при работе с производными. У каждой из масок сумма коэффициентов равна нулю, т. е. эти операторы будут давать нулевой отклик в областях с постоянной яркостью:

маски Собела

$$\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \text{ и } \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix};$$

маски Прюитта

$$\begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix} \text{ и } \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}.$$

Приведенные маски используются для выделения горизонталей и вертикалей. В операторе Собела выводится весовой коэффициент 2 для средних элементов. Такое увеличенное значение уменьшает эффект сглаживания за счет придания большего веса средним точкам.

Лапласиан. Недостатком выделения границ с помощью градиента является получение “широких” границ — шириной в несколько пикселей. Для выделения тонкой границы в один пиксел используется лапласиан двумерной функции $f(x, y)$:

$$\Delta^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}.$$

Поскольку изображение представляется дискретным набором пикселей, оператор Лапласа можно реализовать как процедуру линейной обработки изображения апертурой 3×3 — три строки с номерами 0, 1, 2 и три столбца — 0, 1, 2. Цифровая аппроксимация может быть представлена в виде:

$$\partial^2 f = 4 n_{11} - (n_{01} + n_{10} + n_{12} + n_{21}).$$

Лапласиан применяется для определения, на какой стороне контура — темной или светлой — находится рассматриваемый пиксел.

К недостаткам лапласиана можно отнести очень высокую чувствительность к шумам, возможность появления разрывов в контуре и возникновение ложных контуров. Поэтому вместе с гауссовским сглаживанием очень часто применяется лапласиан. Гауссовское сглаживание и поиск лапласиана можно выполнять одновременно, что сокращает время обработки изображения. Аппроксимирующая маска для получения свертки изображения с лапласианом гауссиана имеет вид [6]:

$$\begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & -2 & -1 & 0 \\ -1 & -2 & 16 & -2 & -1 \\ 0 & -1 & -2 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}.$$

Поскольку взятие второй производной является линейной операцией, свертка изображения с этой маской эквивалентна тому, как если бы изображение сначала сворачивалось с гауссовой сглаживающей функцией, а затем вычислялся лапласиан результата.

Связывание контуров и нахождение границ. Алгоритмы обнаружения контуров обычно дополняются процедурами связывания, чтобы сформировать содержательные контуры из множества контурных точек. Один из простейших подходов к связыванию точек контура состоит в анализе характеристик пикселей в небольшой окрестности каждой точки (x, y) изображения, которая была отмечена как контурная точка. Все точки, являющиеся сходными в соответствии с заранее заданными критериями, связываются и образуют контур, состоящий из пикселей, отвечающих этим критериям. При таком анализе используются следующие два основных параметра для установления сходства пикселей контура:

1) величина отклика оператора градиента, определяющая значение пикселей контура;

2) направление вектора градиента.

Таким образом, пиксел контура, имеющий координаты (x_0, y_0) и расположенный внутри заданной окрестности точки (x, y) , считается сходным по модулю градиента с пикселом (x, y) , если $|\Delta f(x, y) - \Delta f(x_0, y_0)| \leq E$, где E — заданный неотрицательный порог. Направление (угол) вектора градиента задается выражением:

$$\alpha(x, y) = \arctg(g_x/g_y).$$

Пиксел контура, имеющий координаты (x_0, y_0) и расположенный внутри заданной окрестности точки (x, y) считается сходным по направлению градиента с пикселом (x, y) если $|\alpha(x, y) - \alpha(x_0, y_0)| \leq A$,

где A — заданный неотрицательный угловой порог. Отметим, что направление контура в точке (x, y) перпендикулярно направлению вектора градиента в этой точке.

Пиксел в заданной окрестности объединяется с центральным пикселом (x, y) , если выполнены критерии сходства и по величине, и по направлению вектора градиента. Этот процесс повторяется в каждой точке изображения одновременным запоминанием найденных связанных пикселов при движении центра окрестности.

В известном алгоритме Канны для устранения удвоения контура используется гистерезис с двумя пороговыми значениями — высоким и низким. Любой пиксел, который имеет значение больше высокого порога, обозначается как краевой. Затем любые пикселы, которые близко соседствуют с ним и имеют значение выше низкого порога, выбираются как краевые. Для начала движения по контуру необходим высокий порог, а для окончания — низкий.

Нахождение цепочки пикселов с минимальной суммой градиента. Цепочка представляет собой или вертикальную, или горизонтальную непрерывную последовательность пикселов, пересекающую всю картинку. При нахождении вертикальной цепочки каждый пиксел соединяется с пикселом либо непосредственно под ним, либо непосредственно ниже и левее, либо непосредственно ниже и правее. Для этого на основе матрицы градиентов, в которой каждый элемент имеет значение градиента соответствующего пиксела, рассчитывается матрица сумм по алгоритму:

- первая строка аналогична первой строке матрицы градиентов;
- вторая и все последующие строки формируются одинаково — берется значение градиента соответствующего пиксела и складывается с наименьшим значением градиента трех соседних элементов предыдущей строки матрицы.

Алгоритм может быть формализован следующим образом:

$$sum[i, j] = \begin{cases} g[i, j], i = 0, \\ g[i, j] + \min(sum[i - 1, j - 1], sum[i - 1, j], \\ sum[i - 1, j + 1]), i > 0. \end{cases}$$

Цепочки заканчиваются пикселями, расположенными в последней строке. Поиск цепочки начинается с пиксела в последней строке, для которого анализируются три соседних из следующей строки и выбирается минимальный. Затем найденный элемент становится основой для следующего шага и т. д. Найденные цепочки с минимальной суммой могут удаляться из изображения или добавляться в него без изменения размеров основных объектов изображения, так как точки их контуров как имеющие большие значения в цепочки не попадут и важные структурные свойства изображения будут сохранены.

СПИСОК ЛИТЕРАТУРЫ

1. Avidan S., Shamir A. Seam carving for content-aware image resizing. – Mitsubishi Electric Research Labs. [Электронный ресурс], 2007. – URL: <http://www.shaividan.org/papers/imretFinal.pdf> (режим доступа свободный).
2. Perona P., Malik J. Scale-space and edge detection using anisotropic diffusion // IEEE Trans. Pattern Anal. March.Intell. – 1990. – Vol. 12, No. 5. – P. 629–639.
3. Нанно S. Optimal operators in digital image processing: Дис. – Германия, 2000. [Электронный ресурс]. – URL: <http://archiv.ub.uni-heidelberg.de/volltextserver/volltexte/2000/962/>
4. Обработка сигналов и изображений: Image Proc. Toolbox. URL: <http://matlab.exponenta.ru/imageprocess/book/11/fspecial.php>
5. Титов И. О., Емельянов Г. М. Выделение контуров изображения движущегося объекта // Вестник Новгородского гос. ун-та. – 2010. – № 55. – С. 27–31.
6. Гонзалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – С. 812–850.

Статья поступила в редакцию 10.05.2012