

АЛГОРИТМЫ ПОИСКА ПУТИ И РАЗРЕЗА В ГРАФЕ ДЛЯ ПОЛУАВТОМАТИЧЕСКОЙ
СЕГМЕНТАЦИИ МЕДИЦИНСКИХ ИЗОБРАЖЕНИЙ

Сегментация изображений – процесс разбиения множества пикселей исходного изображения на два подмножества: множества пикселей области интереса и множества пикселей вне области интереса. В сфере медицины эта задача часто возникает при анализе изображений компьютерного или магнитно-резонансного томографа, где объектами являются отдельные ткани, опухоли или другие патологии. Решение ее при помощи компьютера существенно улучшает точность и скорость сегментации при диагностике, планировании лечения и проведении хирургических операций. В настоящее время широкое распространение получили методы полуавтоматической сегментации, предполагающие наличие врача, который некоторым образом обозначает точки объекта, затем алгоритм уточняет границы объекта. В данной работе исследуются алгоритмы, основанные на представлении изображения в виде взвешенного графа [1, 2].

Определим взвешенный граф $G = (V, E)$, где V – множество пикселей изображения, E – множество ребер, построенных следующим образом. Пусть $v(x, y)$ – вершина, соответствующая пикселю с координатами x, y . Каждая вершина имеет ребро с вершинами, $v(x + 1, y)$, $v(x, y + 1)$, $v(x - 1, y)$, $v(x, y - 1)$. Обозначим $W_{p,q}$ – вес ребра между вершинами p и q .

При расстановке весов в графе естественным образом можно использовать значения градиента в пикселях изображения. Идея метода в том, чтобы соединить две контрольные точки путем, проходящим по границе области [2]. Для этого веса расставляются таким образом, что путь, проходящий по границе выделяемой области, имеет наименьшую стоимость. Таким образом, задача сводится к поиску кратчайшего пути в графе. Для этого весовая функция задается обратно зависящей от значения градиента, поскольку на границе оно больше, чем в самой области. В ходе исследования была выбрана следующая весовая функция: пусть p, q – два пикселя, I_p – интенсивность пикселя p . Тогда, значение весовой функции вычисляются по формуле.

$$W_{p,q} = \frac{1}{0.1 + \max(\nabla I_p^2, \nabla I_q^2)}$$

Также в процессе работы решались такие задачи, как выбор порядка аппроксимации самого градиента. В итоге применялась формула конечных разностей [3]. Пусть пиксель изображения имеет координаты x, y . Тогда значение градиента вычисляется по формуле:

$$\left| \nabla I_{x,y} \right| = (\max(D_{x,y}^{-x} - D_{x,y}^{+x}, 0)^2 + \max(D_{x,y}^{-y} - D_{x,y}^{+y}, 0)^2)^{1/2}, \text{ где}$$

$$D_{x,y}^{-x} = \frac{I_{x,y} - I_{x-1,y}}{h}; D_{x,y}^{+x} = \frac{I_{x+1,y} - I_{x,y}}{h}; h - \text{ шаг сетки равен } 1, D_{x,y}^{-y} \text{ и } D_{x,y}^{+y} \text{ определяются}$$

аналогично.

К изображению также применялся медианный фильтр для подавления шумов. А для поиска кратчайшего пути применяется алгоритм Дейкстры [4].

Алгоритм, основанный на поиске разреза в графе, принимает на вход множества пикселей объекта и фона, указанные пользователем. Данный метод использует информацию об интенсивности пикселей, и для его работы требуется построение дополнительных ребер в графе следующим образом. Пусть O – множество пикселей объекта, B – множество пикселей фона; выберем $s \in O$ – источник, $t \in B$ – сток. Соединим каждую вершину графа с источником и стоком, вес этих ребер зависит от интенсивности пикселей. Назовем ребра, соединяющие s и t с вершинами из множеств O и B жесткими ограничениями. В полученном графе будем искать минимальный разрез. Подграф, вершины которого достижимы из источника через ребра, не входящие в разрез является множеством пикселей объекта [1]. Задача поиска разреза в графе алгоритмически разрешима за полиномиальное время, сами алгоритмы в данной работе не рассматриваются. Обозначим p – произвольный пиксель изображения, q – пиксель, соседний к p .

Граничные ребра:

$$W_{p,q} = \exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right), \text{ где } \sigma - \text{разница между интенсивностями пикселей,}$$

считающаяся границей.

Жесткие ограничения:

$$W_{b,s} = 0; W_{b,t} = K, b \in B$$

$$W_{o,s} = K; W_{o,t} = 0, o \in O, \text{ где } K > W_{p,q}$$

Ребра с источником и стоком:

$$W_{p,s} = -\ln(Pr(I_p | O))$$

$$W_{p,t} = -\ln(Pr(I_p | B))$$

Здесь $Pr(I_p | O)$ и $Pr(I_p | B)$ – вероятность принадлежности пикселя p объекту или фону соответственно, основанная на интенсивности пикселя.

В качестве функции вероятности использовалась плотность нормального распределения. При использовании данной модели существует риск попадания в сегментированную область всех пикселей, имеющих близкую к интересующему объекту интенсивность. В литературе данная проблема носит название «проблема камуфляжа» [5]. Для её решения в данном алгоритме используется двухэтапная процедура сегментации. На первом этапе изображение сегментируется только с использованием граничных ребер и жестких ограничений. Полученное таким образом начальное приближение используется для поиска параметров функции вероятности и вычисления штрафных коэффициентов для весов ребер с источником и стоком [6]:

$$W_{p,s} = -\ln(Pr(I_p | O)) \exp\left(-\frac{dist(p)^2}{2\sigma_d^2}\right)$$

$$W_{p,t} = -\ln(Pr(I_p | B)) \exp\left(-\frac{dist(p)^2}{2\sigma_d^2}\right), \text{ где } \sigma_d - \text{максимальная допустимая}$$

удаленность пикселей объекта от границы начального приближения.

В полученном графе производится ещё один поиск разреза и его результат возвращается пользователю.

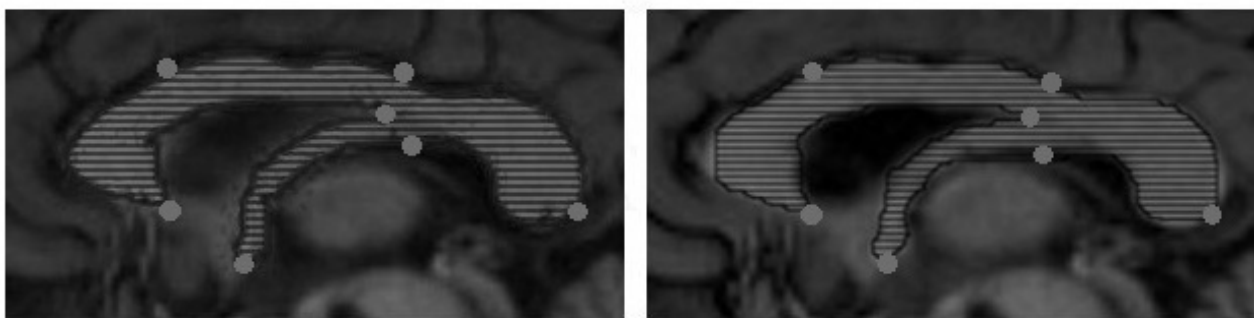


Рис. 1. Результат сегментации: слева – алгоритм поиска пути, описанного в данной статье, справа – инструмент «Умные ножницы» в GIMP. Точки на изображении – выбор пикселей границы пользователем

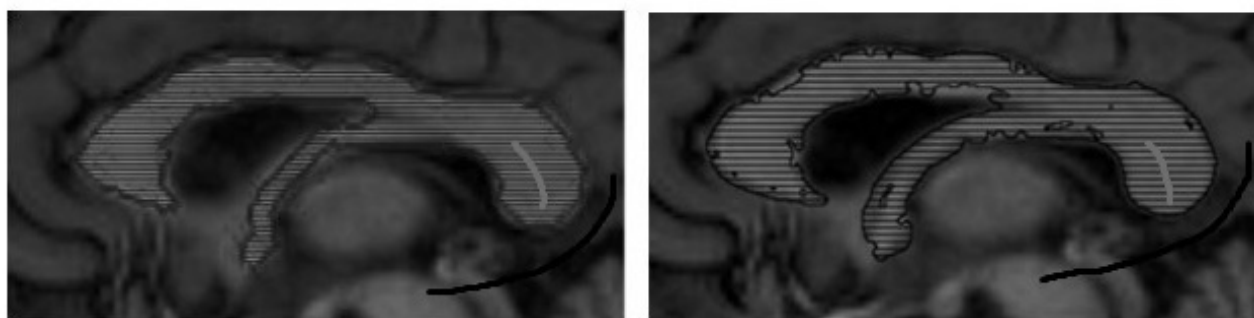


Рис. 2. Результат сегментации: слева – алгоритм поиска разреза, справа – инструмент «Отделение фона» в GIMP. Светлые линии на объекте – пиксели, обозначенные пользователем как объект, черные – фон.

Оба алгоритма были реализованы на языке C++ как независимые модули графического редактора. Алгоритм поиска пути реализован с интерфейсом, предоставляющим возможность выставлять на изображении контрольные точки, которые при помощи данного алгоритма корректируются согласно весам в графе и занимают позиции на границах контрастных объектов. Аналогичный инструмент реализован в графическом редакторе GIMP. Сравнение результатов работы двух инструментов представлено на рис. 1 на примере изображения компьютерного томографа. Перед запуском алгоритма поиска разреза в графе пользователь указывает некоторое количество пикселей, как объекта, так и фона. Аналогичный по интерфейсу инструмент присутствует в GIMP. На рис. 2 представлены результаты их работы на том же примере. В ходе исследования оба представленных в данной работе алгоритма показали лучшие результаты по сравнению с аналогами в GIMP. Инструмент «отделение фона» удобен и прост в применении, так как не требует точного задания исходных данных для работы алгоритма, однако он не всегда позволяет достичь правильного результата в случае нечетких границ. В таких ситуациях инструмент «умные ножницы» позволяет сегментировать объект точнее, но требует больше точек на границе.

ЛИТЕРАТУРА

1. **Boykov Yuri and Funka-Lea Gareth.** Graph Cuts and Efficient N-D Image Segmentation [Журнал]. - [б.м.] : Springer Science & Business Media, 2006 г.
2. **Detlev Stalling and Hans-Christian Hege.** Intelligent Scissors for medical image segmentation [Журнал]. - Berlin : Proceedings of 4th Freiburger Workshop Digitale Bildverarbeitung , 1996 г.

3. **Aubert Gilles and Kornprobst Pierre.** Mathematical problems in image processing: partial differential equations and the calculus of variations [Книга]. - [б.м.] : Springer Science & Business Media, 1996.
4. **Cormen Thomas.** Introduction to algorithms [Книга]. - 1989.
5. **Blake Andrew, Rother Carsten and Hilaire Torr Phillip.** Interactive Image Segmentation Using an Adaptive GMMRF Model [Конференция]. - 2004.
6. **Bieth Marie.** Medical image segmentation using Graph Cut techniques [Отчет]. - [б.м.] : Ecole Polytechnique, 2008.