

Алгоритм нормализованных сечений

Валентин Леонов (гр. M05-8946)

15 декабря 2018 г.

0 Преамбула

Граф $G = (V, E)$ можно разбить на два непересекающихся множества $A, B, A \cup B = V, A \cap B = \emptyset$, удалив ребра, соединяющие эти множества. Степень "непохожести" этих двух частей можно вычислить как общий вес удаленных ребер. В теории графов это называется *весом разреза*:

$$cut(A, B) = \sum_{u \in A, v \in B} w(u, v)$$

Оптимальное разбиение графа на две части - это то разбиение, которое минимизирует вес разреза. Несмотря на экспоненциальное число таких разбиений, нахождение минимального разреза графа является хорошо изученной задачей, и существуют эффективные алгоритмы для ее решения.

Чтобы избежать неестественного "пристания" к отделению небольших наборов вершин, проиллюстрированного на рис. 1, предлагается иная мера "непохожести". Назовем эту меру *нормализованным сечением*:

$$Ncut(A, B) = \frac{cut(A, B)}{assoc(A, V)} + \frac{cut(A, A)}{assoc(B, V)},$$

где $assoc(A, V) = \sum_{u \in A, t \in V} w(u, t)$ - количество ребер между вершинами из A и всеми остальными вершинами графа, а $assoc(B, V)$ определяется аналогично. С таким определением веса, разрез, который отделяет изолированные вершины, не будет иметь маленькое значение $Ncut$.

1 Введение

В данном обзоре описан алгоритм субъективной группировки, который нацелен на извлечение глобальных объектов изображения. Рассматривая задачу группировки как задачу разбиения графа, предложен критерий нормализованного сечения.

Нормализованное сечение - это несмещенная мера разбиения графа на подмножества, которая обладает отличным свойством - минимизация нормализованного сечения непосредственно приводит к максимизации нормализованного объединения, которое является несмещенной мерой для объединения в самих подграфах.

Эффективный алгоритм вычисления минимального нормализованного сечения демонстрирует то, что инвариантное подпространство дает решение нашей задачи.

Вычислительный метод, основанный на этой идее, был разработан и применен для сегментации изображений по яркости, цвету и текстуре.

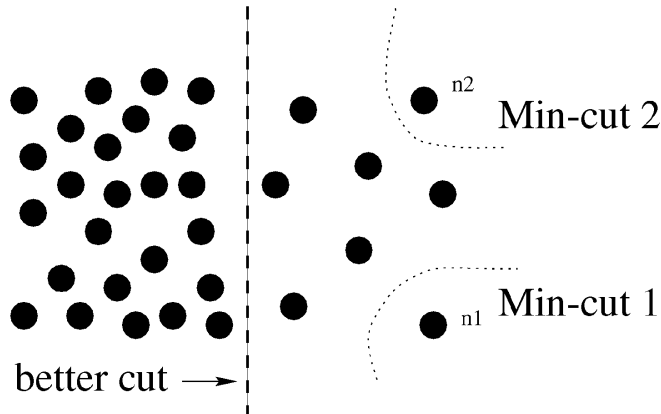


Рис. 1: Пример случая, когда минимальный разрез дает нам неудачное разбиение графа.



Рис. 2: Серое изображение для обработки.

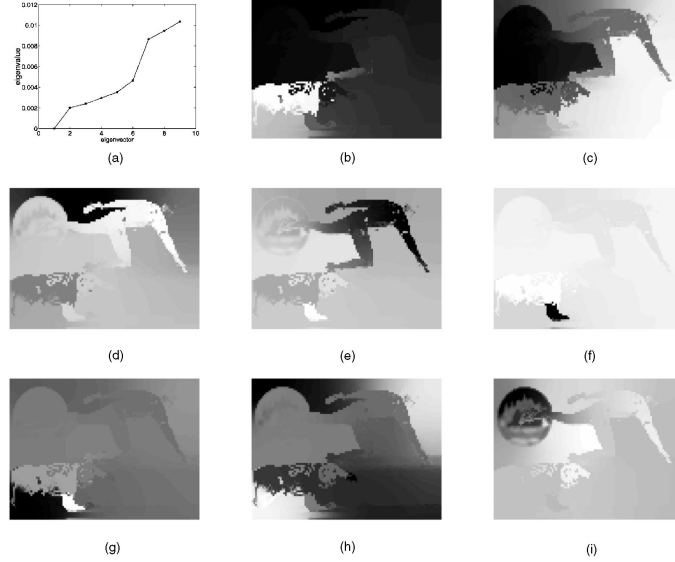


Рис. 3: (а) наименьшие собственные векторы. (b) - (i) собственные векторы, соответствующие наименьшим собственным значениям (со второго по девятое).

2 Алгоритм группировки

2.1 Общее описание

1. Для данного изображения или последовательности изображений строим взвешенный граф $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, реберными весами которого будут меры сходства между двумя вершинами.
2. Решаем систему $(\mathbf{D} - \mathbf{W})\mathbf{x} = \lambda\mathbf{D}\mathbf{x}$ для получения собственных векторов с наименьшими собственными значениями.
3. Используя собственный вектор ([вектор Фидлера](#)) со вторым из минимальных собственных значений, разделяем граф на две части.
4. При необходимости рекурсивно разбиваем получившиеся части.

Подробное описание алгоритма группировки, а также его вычислительную сложность можно проиллюстрировать на следующем примере.

2.2 Описание алгоритма, использующего яркость изображения

На рис. 2 показано изображение, которое мы хотели бы сегментировать. Описание шагов алгоритма:

1. Строим взвешенный граф $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, сопоставив каждый пиксель с вершиной графа и соединив каждую пару пикселей ребром. Вес на этом ребре должен отражать вероятность того, что два пикселя принадлежат одному объекту. Используя только значение яркости пикселей \mathbf{F} и их пространственное расположение \mathbf{X} , мы можем определить вес ребра, соединяющего вершины i и j , как:

$$w_{ij} = e^{\frac{-\|\mathbf{F}(i) - \mathbf{F}(j)\|_2^2}{\sigma_F^2}} * \begin{cases} e^{\frac{-\|\mathbf{X}(i) - \mathbf{X}(j)\|_2^2}{\sigma_X^2}} & \text{если } \|\mathbf{X}(i) - \mathbf{X}(j)\|_2 < r \\ 0 & \text{в противном случае.} \end{cases}$$

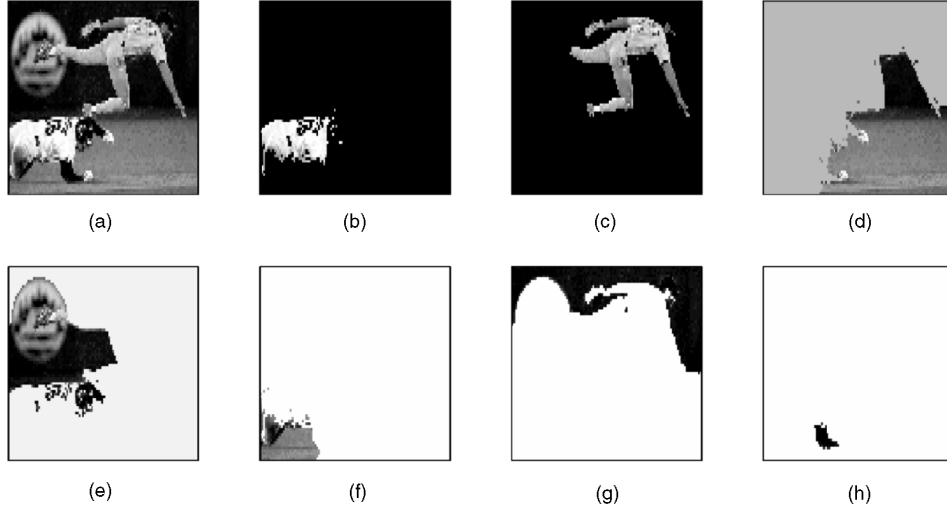


Рис. 4: (a) исходное изображение размером 80×100 . (b) - (h) сегменты изображения, для значения $Ncut$ менее 0,04. Значения параметров: $\sigma_I = 0.1, \sigma_X = 4.0, r = 5$.

2. Сводим систему

$$(\mathbf{D} - \mathbf{W})\mathbf{y} = \lambda \mathbf{D}\mathbf{y}$$

к стандартной задаче вычисления собственных значений

$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}}\mathbf{x} = \lambda \mathbf{x}$$

и получаем собственные вектора с наименьшими собственными значениями. Решение задачи вычисления собственных значений для всех собственных векторов требует $\mathcal{O}(n^3)$ операций, где n - количество вершин в графе. Это слишком медленно для задачи сегментации изображения, где n - это количество пикселей в изображении. К счастью, наше разбиение графа имеет следующие свойства:

- (a) Подграфы чаще всего связаны только локально, и полученные системы довольно разрежены.
- (b) Для разбиения графа достаточно всего нескольких собственных векторов.
- (c) Требуется низкая точность для собственных векторов, часто нужен лишь знак.

Данные свойства позволяют воспользоваться методом Ланцоша. Временная сложность алгоритма Ланцоша составляет $\mathcal{O}(mn) + \mathcal{O}(m\mathcal{M}(n))$, где m - максимальное количество требуемых умножений матрицы на вектор, а $\mathcal{M}(n)$ - стоимость вычисления матрицы $\mathbf{A}\mathbf{x}$, где $\mathbf{A} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}}$. А поскольку матрица весов \mathbf{W} и матрица \mathbf{A} разрежены, умножение происходит за $\mathcal{O}(n)$.

Чтобы понять, почему это так, рассмотрим стоимость скалярного произведения одной строки \mathbf{A} с вектором \mathbf{x} . Пусть $\mathbf{y}_i = \mathbf{A}_i \cdot \mathbf{x} = \sum_j \mathbf{A}_{ij}\mathbf{x}_j$. Для фиксированного i , \mathbf{A}_{ij} не равно нулю, только если вершина j находится в пространственной окрестности вершины i . Следовательно, для вычисления $\mathbf{A}_i \cdot \mathbf{x}$ требуется фиксированное число операций, а общая стоимость вычисления $\mathbf{A}\mathbf{x}$ составляет $\mathcal{O}(n)$.

Внеасимптотическая константа определяется алгебраической связностью графа. Оказывается, мы можем существенно сократить количество ребер от каждой вершины к смежным с ней путем случайного удаления этих ребер в некоторой окрестности вершины. Опытным путем было обнаружено, что можно удалить до 90% всех ребер в большой окрестности вершины, не влияя при этом на собственные векторы системы.

В конечном итоге, каждое из произведений матрицы на вектор стоит $\mathcal{O}(n)$ операций с небольшой константой. Число m зависит от многих факторов. В экспериментах по сегментации изображений оказывалось, что m обычно меньше $\mathcal{O}(n^{\frac{1}{2}})$.

3. После вычисления собственных векторов мы можем разбить граф на две части, используя второй из наименьших собственных векторов. В идеальном мире собственный вектор будет состоять из двух целых значений, а их знаки точно скажут нам, как разбить граф. Однако наши собственные векторы могут принимать вещественные значения, и нам нужно выбрать точку разделения, чтобы разделить граф на две части. Есть много разных способов выбора такой точки. Можно взять 0 или медиану в качестве точки разделения, или можно искать ее так, чтобы результирующее разделение имело наилучшее значение $Ncut(\mathbf{A}, \mathbf{B})$.
4. После того, как граф разбит на две части, мы можем рекурсивно запустить наш алгоритм на этих частях. Или, что эквивалентно, мы могли бы воспользоваться особыми свойствами других наименьших собственных векторов, чтобы разделить граф с их помощью. Рекурсия останавливается, как только значение $Ncut$ превышает определенный предел.

Список литературы

- [1] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 731–, Washington, DC, USA, 1997. IEEE Computer Society.
- [2] Jianbo Shi and Jitendra Malik. Motion segmentation and tracking using normalized cuts. In *Proceedings of the Sixth International Conference on Computer Vision*, ICCV '98, pages 1154–, Washington, DC, USA, 1998. IEEE Computer Society.
- [3] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, August 2000.