

# Автоматическая настройка параметров алгоритма сегментации изображений Quick Shift

Максим Колосовский, Елена Крючкова  
Факультет информационных технологий

Алтайский государственный технический университет им. И.И. Ползунова, Барнаул, Россия  
maxim.astu@gmail.com kruchkova\_elena@mail.ru

## Аннотация

Качество работы алгоритмов пересегментации принципиально зависит от свойств изображения и требует настройки параметров под конкретное изображение. При использовании неподходящих параметров пересегментация не сохранит границы объектов изображения или построит слишком много мелких суперпикселей, что снизит компактность такого представления. В данной статье представлены исследования, направленные на выбор оптимальных параметров алгоритма сегментации Quick Shift. Для построения системы адаптации этого алгоритма, мы исследовали настройку отдельных параметров. Наш подход отличается от аналогичных тем, что основан на использовании метрик изображения, требующих принципиально меньше вычислительных затрат, чем многократное выполнение сегментаций, характерное для большинства других подходов. Полученные выводы позволяют перейти к построению единой системы настройки параметров Quick Shift.

**Ключевые слова:** пересегментация, семантическая сегментация, Quick Shift, настройка параметров.

## 1. ВВЕДЕНИЕ

Одной из главных сложностей задачи семантической сегментации произвольного изображения является зависимость оптимальных значений параметров алгоритма от свойств изображений. Например, изображения, где объекты и фон хорошо различимы, и изображения, где объекты и фон похожи по цвету, требуют различных значений параметров, отвечающих за определение разницы цвета граничащих объектов. Другой пример – это размер объектов. Области одного и того же размера на одном изображении могут являться интересующими нас объектами, а на другом – деталями, которые не существенны для семантики изображения.

В данной работе мы на примере алгоритма Quick Shift [6, 7] представляем подход к настройке параметров алгоритма пересегментации. Для построения системы адаптации алгоритма мы исследуем настройку отдельных параметров при помощи вычисления сконструированных нами метрик, определяющих необходимые свойства изображения. Следующие три секции посвящены трем параметрам алгоритма Quick Shift: компактность цветов суперпикселей (секция 2), расстояние между пикселями в кластере (секция 3), ширина окна оценки плотности (секция 4).

В работе использовалась база изображений PASCAL VOC 2007 [1].

## 2. КОМПАКТНОСТЬ ЦВЕТОВ СУПЕРПИКСЕЛЕЙ

В рассматриваемой реализации алгоритма Quick Shift в качестве признаков пикселя используется не только значение цвета пикселя, но и его координаты на изображении. Благодаря этому кластеры в таком пространстве будут соответствовать группам пикселей, находящихся рядом на изображении. Без учета пространственного положения части одного кластера могут быть «разбросаны» по всему изображению, что порождает множество очень мелких суперпикселей и не соответствует цели пересегментации.



**Рис 1.** Исходное изображение и его пересегментации при различных значениях параметра цветовой компактности суперпикселей.

Данная реализация позволяет регулировать компактность цветов суперпикселя за счет параметра *ratio* в условии объединения пикселей в один кластер:

$$(da^2 + db^2 + dc^2) * ratio + dx^2 + dy^2 \leq maxdist^2 \quad (1)$$

где *da*, *db*, *dc* – расстояние между пикселями по компонентам цвета (их может быть и меньше трех), *ratio* – параметр компактности цветов суперпикселя, *dx*, *dy* – разность координат сравниваемых пикселей, *maxdist* – параметр, задающий максимальное расстояние между пикселями, при котором их можно объединить в один кластер. Чем больше значение параметра *ratio*, тем более похожие по цвету пиксели нужно объединять в один суперпиксель, тем, соответственно, более мелкие суперпиксели получатся и тем менее компактно будет представление изображения. С другой стороны, при слишком малом значении параметра *ratio* в один суперпиксель будут попадать не похожие по цвету пиксели, что, как и излишнее количество регионов, не отвечает целям сегментации изображения. Получается, что нужен компромисс в определении цветовой компактности суперпикселей (рисунок 1).

Основная проблема настройки этого параметра заключается в том, что компромиссное значение для разных изображений может быть очень различно (рисунок 2). Авторами были выявлены некоторые зависимости оптимального значения параметра *ratio* от свойств изображения [8]:

- чем более похожи цвета граничащих объектов, тем значение *ratio* должно быть больше;
- чем больше резко изменчивой текстуры (ветки деревьев, трава, провода, мелкие надписи и т.п.), тем больше при

увеличении параметра появляется очень мелких суперпикселей. Если все текстуры гладкие, то увеличение *ratio* слабо изменит сегментацию. Таким образом, отсутствие изменчивой текстуры позволяет задавать меньшее значение параметра;

- чем мельче размеры искоемых объектов, тем *ratio* должно быть больше.



**Рис 2.** Разные изображения по разному сегментируются при увеличении параметра цветовой компактности

Эти критерии демонстрируют зависимости, но не позволяют автоматически вычислять оптимальное значение параметра для конкретного изображения. В этой работе мы попытались решить эту задачу. Мы нашли метрику изображения, которая взаимосвязана с оптимальным значением параметра *ratio*, после чего построили график этой зависимости метрики от значения параметра.

Было замечено, что оптимальное значение *ratio*, при котором не образуется очень много мелких суперпикселей, зависит от «изменчивости» текстуры. Например, если текстура представляет собой комбинацию пикселей очень разных цветов, то при большом значении *ratio* текстура «распадается» во множество мелких суперпикселей. Если же текстуры на изображении гладкие, то можно использовать гораздо большее значение *ratio* и получить более компактное представление изображения. Изменчивость текстуры *TextureContast* мы измеряем как среднюю разность по цвету между всеми парами соседних пикселей, имеющими общую сторону:

$$HorizContrast = \frac{1}{h * (w - 1)} \sum_{i=1}^h \sum_{j=1}^{w-1} dist(I_{i,j}, I_{i,j+1})$$

$$VertContrast = \frac{1}{(h - 1) * w} \sum_{j=1}^w \sum_{i=1}^{h-1} dist(I_{i,j}, I_{i+1,j})$$

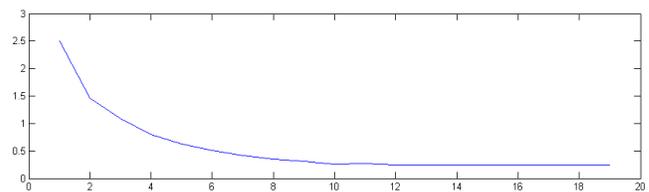
$$TextureContast = HorizContrast + VertContrast$$

где *h*, *w* – высота и ширина изображения в пикселях, *I<sub>ij</sub>* – цвет пикселя в позиции (*i*, *j*), *dist(a,b)* – расстояние между цветами *a* и *b* в пространстве Luv. На рисунке 2 представлены два изображения с наименьшей и наибольшими значениями метрик среди коллекции PASCAL VOC.

Эксперименты подтвердили наличие обратно пропорциональной зависимости оптимального значения параметра от значения предложенной метрики. Чтобы формализовать зависимость параметра от метрики, мы ввели определение «оптимальности» значения параметра, основанное на эмпирических наблюдениях. Оптимальным значением параметра *ratio* будем считать максимальное

значение, при котором суперпиксели размером менее 50 пикселей покрывает не более 2% площади изображения.

На изображениях коллекции PASCAL VOC 2007 выполнены пересегментации с различными значениями параметра *ratio* (от 0.1 до 2.5) и выбрано оптимальное из них. Для всех изображений также вычислена представленная метрика. Изображения сгруппированы по значению метрики. Для каждой группы изображений подсчитано среднее оптимального значения параметра. График, получившейся зависимости, представлен на рисунке 3.



**Рис 3.** Зависимость оптимального значения цветовой компактности от значения метрики

Таким образом, мы получили метод автоматической адаптации параметра цветовой компактности суперпикселя в алгоритме Quick Shift. Этот метод позволяет по представленному изображению оценить оптимальное значение параметра алгоритма. Заметим, что оценка производится без подгонки параметров или какого-либо предварительного обучения по прецедентам за исключением использования графика зависимости, представленного на рисунке 3.

### 3. РАССТОЯНИЕ МЕЖДУ ПИКСЕЛЯМИ В КЛАСТЕРЕ

В данном разделе продемонстрирован способ оценки оптимального значения параметра *maxdist* при помощи обучения на тестовых изображениях, имеющих эталонную семантическую сегментацию.

Алгоритм Quick Shift решает задачу сегментации как задачу кластеризации. В отличие от алгоритма кластеризации *k*-средних, требующего задания количества кластеров, Quick Shift требует максимального расстояния между элементами кластера. В этом разделе мы рассмотрим подход к оценке этого расстояния.

Quick Shift при кластеризации объединяет все пиксели изображения в одно дерево. В этом дереве родитель пикселя – это ближайший к нему в пространстве признаков (т.е. цвет + расположение) пиксель с большей плотностью. После этого удаляются все ребра больше некоторого порога – *maxdist*. Получившийся лес является результатом кластеризации пикселей, т.е. набором суперпикселей. Ключевое значение в кластеризации занимает выбор этого порога. Чем больше порог, тем более непохожие по цвету или дальше расположенные пиксели будут объединяться в один суперпиксель, тем, следовательно, выше вероятность объединения пикселей разных объектов. С другой стороны, если порог будет слишком мал, то суперпиксели будут чересчур мелкими. Цель подбора этого параметра – выбрать значение, при котором будет достаточно точно определяться граница между объектами при минимально возможном числе суперпикселей.

Как и в случаях с другими параметрами, оптимальное значение сильно зависит от содержания изображения. Если объект(ы) и фон имеют контрастирующие цвета на своих границах и граница не размыта, то порог можно увеличить. Если же цвета очень схожи, то придется пожертвовать компактностью пересегментации и уменьшить значение параметра *maxdist*. На рисунке 4 видно, что для одних изображений увеличение параметра *maxdist* незначительно ухудшает точность выделения границ, в то время как для других требуется очень мелкая сегментация из-за слабого контраста фона и объекта.



Рис 4. Влияние на различные изображения параметра максимального расстояния между пикселями

Оценка метрики происходит следующим образом. Метрика вычисляется по эталонной семантической сегментации. Напомним, что на эталоне пиксели отмечаются как принадлежащие к какому-либо объекту или фону. В коллекции PASCAL VOC также отмечаются граничные пиксели, семантическое значение которых не определено.



Рис 5. Несколько изображений с минимальным (верхний ряд) и максимальным (нижний ряд) значением метрики среди изображений коллекции PASCAL VOC 2007

Рассмотрим *i*-ую строку эталонной семантической сегментации. Будем сканировать эту строку слева направо. Нас интересуют переходы от одного объекта к другому. Пусть мы рассматриваем переход от пикселей объекта А к пикселям объекта В. Пусть крайний правый пиксель объекта А находится в столбце *j*, а крайний левый пиксель объекта В в столбце *k*.

$$Sem_{i,j} = Object A$$

$$Sem_{i,k} = Object B$$

$$Sem_{i,t} = BORDER, j < t < k$$

где  $Sem_{ij}$  – семантическая метка пикселя, расположенного на пересечении *i*-ой строки и *j*-го столбца, в частности, метка «BORDER» означает граничный пиксель.

Вычислим для этого перехода следующую величину:

$$T = \sqrt{ratio * dist(I_{i,j}, I_{i,k})^2 + (k - j)^2},$$

где *ratio* – параметр цветовой компактности,  $I_{ij}$  – цвет пикселя,  $dist(C1, C2)$  – расстояние между цветами *C1* и *C2*.

Эта величина равняется расстоянию между пикселями (*i,j*) и (*i,k*) в используемом пространстве признаков (формула (1)).

Подсчитаем среднее значение этой величины по всем переходам во всех строках, а также подобные переходы при сканировании не только строк, но и столбцов. Отметим, что если группа граничных пикселей шириной более 10 пикселей, то такой переход не учитывается, т.е.  $j < k \leq j + 10$ .

На рисунке 5 представлены изображения, имеющие минимальное и максимальное значение метрики среди изображений коллекции PASCAL VOC 2007. На изображении с минимальным значением трудно рассмотреть объект, так как контраст объекта и фона очень небольшой. На изображении, где метрика максимальна, напротив, объект хорошо отличим от фона.



Рис 6. Пример пересегментации при значении параметра, равного половине значения рассматриваемой метрики (3-ий столбец), а также пересегментации при большем и меньшем значении параметра (2-ой и 4-ый столбцы)

Было отмечено, что оптимальное значение параметра достигается при значении половины значения метрики. Дело в том, что эта метрика вычисляет расстояние в пространстве признаков между пикселями, находящимися по разные стороны от границы, поэтому для разделения пикселей границы примерно посередине нужно брать порог, равный половине этой метрики. Примеры пересегментации со значением параметра, равным половине значения метрики, представлены на рисунке 6. Таким образом, мы получили инструмент для определения оптимального значения параметра *maxdist* по коллекции изображений, который позволяет оценить средний уровень контраста между объектами и фоном и использовать эту информацию при обработке новых изображений.

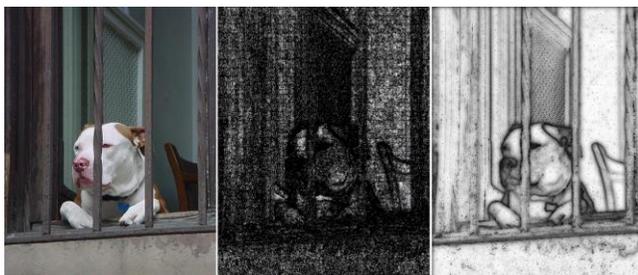
#### 4. ШИРИНА ОКНА ОЦЕНКИ ПЛОТНОСТИ

Кластеризация пикселей в алгоритме Quick Shift основывается на оценке плотности расположения пикселей в пространстве признаков: место где, плотность достигает локального максимума, считается центром кластера. Плотность точки можно оценить как количество точек, попавших в окрестность заданного радиуса (*окно Парзена*). Однако чтобы учитывать не только количество, но и расстояние до этих точек, вводят специальную функцию, называемую *ядром*, которая делает вклад более удаленных точек меньше. Обычно в качестве ядра используют Гауссово распределение.

Чем больше размер окна, тем функция плотности будет более гладкой, а значит, тем более крупные кластеры будут получаться. С другой стороны, слишком узкое окно построит функцию плотности со слишком большим количеством экстремумов. Напомним, что хорошая пересегментация – это компромисс между компактностью представления и точностью выделения контуров объектов.

На рисунке 7 представлено изображение и значения плотностей при радиусе окна, равного 1 и 12 пикселям. Плотности изображены в виде градаций серого, где белый цвет соответствует максимальному значению плотности, а

черный – минимальному. При узком окне значение плотности большинства пикселей ближе к минимальному, а при широком наоборот. Это связано с тем, что функция плотности представляет собой сумму функций ядер, а, как известно, ядро с узким окном – это резкий пик, а широкое ядро – очень широкий «колокол», где значение, близкое максимальному, распространяется далеко от центра. Таким образом, функция суммы таких «колоколов» имеет много значений близких к максимуму.



**Рис 7.** Исходное изображение и значение плотности при различных размерах окна, равных 1 (в центре) и 12 (справа)

В экспериментах радиус окна Парзена варьировался в пределах от 1 до 12 пикселей. Меньшее значение ширины окна бессмысленно, так как в окно будет попадать лишь сам пиксель, а окрестность не будет учитываться. Ограничение сверху объясняется существенным увеличением вычислительных затрат при увеличении ширины окна. Окно радиусом  $r$  пикселей способно строить отдельные суперпиксели для деталей изображения размером  $2r$  и более. Поэтому ширина окна, как правило, гораздо меньше размера объектов, а значит, оптимальная ширина окна зависит не от размера объектов, а от частей объекта, которые хотелось бы выделять как отдельные суперпиксели. Кроме того, на точность выделения границ влияет то, как расположены области внутри объекта. Например, если на границе объекта есть область небольшого размера, похожая по цвету на фон, то более широкое окно не выделит для этой области отдельного суперпикселя, а присоединит ее к фону. Таким образом, для определения оптимальной ширины окна будет недостаточно даже эталонной семантической сегментации, т.к. ошибки в выделении контуров объектов зависят от конфигурации отдельных частей объектов.

В общем случае увеличение ширины окна должно улучшать сегментацию, потому как учитывается более широкая окрестность пикселя, а пропуск мелких деталей является негативным последствием объединения цветового и пространственного расстояния между пикселями (формула (1)). Например, возможно объединение в один кластер далеких по цвету, но близко расположенных пикселей, вследствие чего в один суперпиксель могут попадать пиксели очень разных цветов. Другим часто наблюдаемым недостатком алгоритма Quick Shift являются ошибки на размытых границах объектов. Если даже цвета объектов существенно различаются, то размытый переход может объединить части этих объектов в один суперпиксель, так как расстояние между соседними пикселями в этом переходе не слишком большое.

Таким образом, этот параметр нуждается скорее не автоматической настройке, а в модификации алгоритма Quick Shift, исправляющей дефекты, связанные с неподходящим значением этого параметра.

## 5. ОБЗОР ПОДОБНЫХ РАБОТ

Автоматическая настройка параметров алгоритмов сегментации является активно исследуемой областью в настоящее время. В ней применяется крайне широкий спектр принципиально различных подходов, среди которых сравнение нескольких сегментаций одного изображения, построенных с разными параметрами [2], многократный прогон алгоритма сегментации с настройкой параметров по предыдущему прогону [5], сложная калибровка одних параметров при фиксировании других [3].

Несмотря на такое большое разнообразие применяемых подходов, наш алгоритм отличается тем, что мы настраиваем его при помощи вычисления метрик изображения, которые требуют принципиально меньших вычислительных затрат, чем построение нескольких сегментаций. Кроме того, мы ставим цель настраивать сразу все параметры алгоритма, а не только ширину окна, что характерно для работ по настройке алгоритмов на основе оценки плотности [4].

## 6. ЗАКЛЮЧЕНИЕ

В данной статье мы исследовали настройку отдельных параметров алгоритма сегментации изображений на примере алгоритма Quick Shift. Полученные выводы по настройке отдельных параметров позволяют перейти к построению единой системы настройки параметров Quick Shift.

## 7. ССЫЛКИ

- [1] Everingham M., van Gool L., Williams C., Winn J., Zisserman. A. The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results // International Journal of Computer Vision (IJCV). June, 2010. Vol. 88, No. 2. P. 303-338.
- [2] Franek L., Jiang X. Adaptive Parameter Selection for Image Segmentation Based on Similarity Estimation of Multiple Segmenters. // 10th Asian Conference on Computer Vision, Queenstown, New Zealand, November 8-12, 2010. pp 697-708.
- [3] Franek L., Jiang X. Alternating Scheme for Supervised Parameter Learning with Application to Image Segmentation. // 14th International Conference, CAIP 2011, Seville, Spain, August 29-31, 2011, pp 118-125.
- [4] Gilani S., Rao N. Data Driven Bandwidth for Medoid Shift Algorithm // International Conference, Santander, Spain, June 20-23, 2011. pp 534-546.
- [5] Khan A., Mikut R., Schweitzer B., Weiss C., Reischl M. Automatic Tuning of Image Segmentation Parameters by Means of Fuzzy Feature Evaluation // Synergies of Soft Computing and Statistics for Intelligent Data Analysis. Vol 109. pp 459-467.
- [6] Vedaldi A., Soatto S. Quick Shift and Kernel Methods for Mode Seeking // European Conference on Computer Vision (ECCV). Marseille, France. 12-18 October, 2008. pp. 705-718.
- [7] Vedaldi A., Fulkerson B. VLFeat: An open and portable library of computer vision algorithms // 18th Annual ACM International Conference on Multimedia (ACM MM). Firenze, Italy. 25-29 October, 2010. P. 1469-1472.
- [8] Колосовский М., Крючкова Е. Настройка параметров алгоритма сегментации изображений QuickShift // Программная инженерия. №5, 2013, сс. 11-20.

## **Об авторах**

Максим Колосовский – аспирант ФИТ АлтГТУ им. И.И. Ползунова. Его адрес: [maxim.astu@gmail.com](mailto:maxim.astu@gmail.com)

Елена Крючкова – профессор ФИТ АлтГТУ им. И.И. Ползунова. Ее адрес: [kruchkova\\_elena@mail.ru](mailto:kruchkova_elena@mail.ru).