

# Технология текстурной сегментации изображений на основании марковских случайных полей и решения (max,+) задач

И. В. Ковтун

## 1. Введение

Текстуры являются важной характеристикой изображений естественных объектов и важным элементом зрительного восприятия. Машинному анализу текстурных изображений посвящено значительное количество исследований (см. список литературы в [1]). В круге этих исследований существенную часть занимает проблема сегментации изображений по текстурным признакам.

Как показано в [2-6] марковские случайные поля могут использоваться для моделирования текстур. Возможные постановки задачи сегментации рассмотрены в [7]. Эти постановки базируются на модели текстурного изображения, основанной на марковских случайных полях. В данной работе предлагается технология текстурной сегментации, объединяющая в себе описание текстур при помощи марковских случайных полей и последние достижения в решении (max,+) задач [8-11].

## 2. Формулировка задачи текстурной сегментации

Пусть задано поле зрения  $T = \{(i, j) | 0 \leq i < I, 0 \leq j < J\}$ , алфавит  $L = \{1, 2, \dots, l\}$  меток и структура соседства  $\mathfrak{S} \subset T \times T$  такая, что соседними с пикселем  $(i, j)$  являются пиксели  $(i-1, j)$ ,  $(i+1, j)$ ,  $(i, j-1)$  и  $(i, j+1)$ .

Разметкой поля зрения  $T$  будем называть функцию  $k_T : T \rightarrow L$ . Обозначим через  $k_t$  значения функции  $k_T$  в пикселе  $t$ .

Пусть  $g_{t,t'}(r, r')$  - числа, заданные для каждой пары  $\{t, t'\} \in \mathfrak{S}$  соседних пикселей и для каждой пары  $\{r, r'\} \in L \times L$  меток. Эти числа не зависят от входного изображения и определяют априорную информацию о возможных разметках.

Пусть  $q_t(r)$  - действительные числа, заданные для каждого пиксела  $t \in T$  и каждой метки  $r \in L$ . Эти числа зависят от входного изображения: число  $q_t(r)$  определяет меру схожести изображения в окрестности пиксела  $t \in T$  и  $r$ -той текстуры.

По этим данным необходимо найти оптимальную разметку, обозначаемую в дальнейшем  $k_T^*$ .

$$k_T^* = \arg \max_{k_T} \left( \sum_{t \in T} q_t(k_t) + \sum_{\{t, t'\} \in \mathfrak{S}} g_{t,t'}(k_t, k_{t'}) \right). \quad (1)$$

Задачи такого типа называются (max,+) задачами.

Далее приводится метод получения величин  $g_{t,t'}(r, r')$  и  $q_t(r)$ , а также метод решения задачи (1).

## 3. Марковские случайные поля, как модели текстур

Пусть  $X$  - конечное множество сигналов. Функцию  $x_T : T \rightarrow X$  будем называть изображением. Обозначим через  $x_t$  значение функции  $x_T$  в пикселе  $t \in T$ .

Предположим, что изображение  $x_T$   $r$ -того типа текстур является марковским случайным изображением.

Марковость определяет тот факт, что распределение вероятностей изображений  $x_T$  имеет вид

$$p(x_T/r) = z_r \cdot \exp \sum_{t \in T} \sum_{\tau \in T_0^r} \varphi_\tau^r(x_t, x_{t+\tau}), \quad (2)$$

где

- $z_r$  - нормализующий множитель, который обеспечивает равенство  $\sum_{x_T: T \rightarrow X} p(x_T/r) = 1$ .
- $T_0^r$  задает однородную структуру соседних пикселей  $\mathfrak{Z}^r$  на поле зрения  $T$ :  

$$\{t, t'\} \in \mathfrak{Z}^r \iff (t' - t) \in T_0^r.$$
- функции  $\varphi_\tau^r: X \times X \rightarrow R$  называются гиббсовскими потенциалами.

Алгоритм оценки параметров марковского случайного поля (структуры соседних пикселей и соответствующих гиббсовских потенциалов) для одной текстуры предложен в работах [2-6].

#### 4. Подсчет величин для задачи сегментации

Положим

$$g_{t,t'}(r, r') = \begin{cases} C \geq 0 & r = r' \\ 0 & r \neq r'. \end{cases} \quad (3)$$

Формула (3) выражает априорное предположение о том, что в соседних пикселях более вероятны одинаковые метки, чем разные.

Далее приводится описание метода подсчета качеств  $q_t(r)$  и его обоснование.

Покажем, что знание структуры поля зрения  $\mathfrak{Z}^r$  и гиббсовских потенциалов  $\varphi_\tau^r$  не является достаточным для решения задач текстурной сегментации.

Пусть структура поля зрения  $\mathfrak{Z}^r$  и числа  $\varphi_\tau^r$  заданы для каждой текстуры  $r \in L$ . Пусть некоторый прямоугольный участок  $T'$  поля зрения размером, скажем,  $10 \times 10$  пикселей вырезан из изображения  $x_T: T \rightarrow X$ . Предположим известным, что весь участок принадлежит только одной текстуре  $x_{T'}: T' \rightarrow X$ . Неизвестно, однако, какой именно текстуре. Если тип текстуры должен быть определен, и каждая из текстур имеет одинаковую априорную вероятность, то необходимо подсчитать апостериорную вероятность каждой текстуры в квадрате, а именно, числа

$$p(r/x_{T'}) = \frac{p(x_{T'}/r)}{\sum_{r'=1}^l p(x_{T'}/r')}. \quad (4)$$

Таким образом, речь идет о подсчете  $l$  чисел. Структуры  $\mathfrak{Z}^r$  и потенциалы  $\varphi_\tau^r$  позволяют вычислить  $l$  величин

$$f_r(x_{T'}) = \sum_{t \in T'} \sum_{\tau \in T_0^r} \varphi_\tau^r(x_t, x_{t+\tau}), r \in L. \quad (5)$$

Подставив в (4) формулы (2) и (5) получим

$$p(r/x_{T'}) = \frac{z_r \cdot \exp f_r(x_{T'})}{\sum_{r'=1}^l (z_{r'} \cdot f_{r'}(x_{T'}))}$$

Последнее равенство включает неизвестные коэффициенты  $z_r$ . Проблема в том, что  $l$  величин  $p(r/x_{T'})$ ,  $r \in L$ , известны лишь с точностью до  $l$  чисел  $z_r$ ,  $r \in L$ . В этом случае отсутствие знаний коэффициентов  $z_r$  приводит к тому, что апостериорные вероятности  $p(r/x_{T'})$  абсолютно неизвестны. Грубо говоря, они могут быть любыми.

Проблема, связанная с коэффициентами  $z_r$ , очень серьезная.

Во-первых, они не могут быть подсчитаны непосредственно, поскольку задаются формулой

$$z_r^{-1} = \sum_{x_T: T \rightarrow X} \exp \sum_{t \in T'} \sum_{\tau \in T_0^r} \varphi_\tau^r(x_t, x_{t+\tau}),$$

то есть их подсчет требует суммирования неимоверно большого числа слагаемых.

Во-вторых, нет оснований неосмотрительно предполагать, что влияние этих коэффициентов

можно игнорировать, то есть предполагать, что они почти равны. Они могут существенно отличаться один от другого, например  $z_1 = 10^4$  и  $z_2 = 10^{-4}$ .

Таким образом, числа  $f_r(x_{T_r})$  не дают возможности непосредственно подсчитать апостериорные вероятности  $p(r/x_{T_r})$ . Далее мы укажем способ, позволяющий оценить апостериорные вероятности  $p(r/x_{T_r})$ . Числа  $f_r(x_{T_r})$  могут пониматься, как отображение изображения  $x_{T_r}$  в новое  $l$ -мерное пространство признаков, где ему соответствует  $l$ -мерный вектор  $f(x_{T_r})$ , компонентами которого являются числа  $f_r(x_{T_r})$ ,  $r \in L$ . Будем называть это пространство признаков  $f$ -пространством. В  $f$ -пространстве оценим условные вероятности  $p(f/r)$  для каждого вектора  $f$  на основании примеров изображений  $r$ -той текстуры и используем их для подсчета апостериорной вероятности текстур. Оценка условных вероятностей  $p(f/r)$  может быть приближена при помощи  $l$ -мерного нормального гауссовского распределения вероятностей. Это предположение является почти эвристическим. Оно следовало бы из закона больших чисел, так как каждая из компонент вектора  $f(x_{T_r})$  является суммой случайных величин. Однако эти случайные величины не являются в нашем случае независимыми. Несмотря на это, данное предположение было экспериментально проверено и использовано в качестве основы для следующего алгоритма оценки.

Исходными данными для алгоритма оценки являются:

- а)  $l$  тестовых изображений  $x_T^r : T \rightarrow X$ ,  $r \in L$ ;
- б) структуры  $T_0^r$  и гиббсовские потенциалы  $\phi_r^t$  для каждой текстуры  $r$  и для каждого  $t \in T_0^r$ ;
- в) размер окошка  $\Delta$  (его смысл станет понятен позже).

Технология оценки условных вероятностей  $p(f/r)$  состоит в следующем.

- Каждое тестовое изображение  $x_T^r$ ,  $r \in L$ , сканируется окошком размером  $\Delta \times \Delta$  и для каждого его положения фиксируется изображение внутри этого окошка. В результате получаем множество  $\tilde{X}$  изображений, которые разделяются на  $l$  подмножеств  $\tilde{X}_1, \tilde{X}_2, \dots, \tilde{X}_l$ , где подмножество  $\tilde{X}_r$  состоит из всех фрагментов изображения  $r$ -той текстуры. Каждый из этих фрагментов определен на поле зрения размером  $\Delta \times \Delta$ . Обозначим это поле зрения  $T_\Delta$ .
- Для каждого изображения  $x_{T_\Delta} \in \tilde{X}$ , независимо от того, какой текстуре оно принадлежит, подсчитывается схожесть  $f_r(x_{T_\Delta})$  с каждой из  $l$  текстур. Массив чисел  $f_r(x_{T_\Delta})$ ,  $r \in L$ , интерпретируется как  $l$ -мерный вектор  $f(x_{T_\Delta})$ . Таким образом, получается множество  $F$   $l$ -мерных векторов, которое разделяется на  $l$  подмножеств  $F_1, F_2, \dots, F_l$ . А именно, вектор  $f$  принадлежит подмножеству  $F_r$ , если он был получен по изображению  $x_{T_\Delta}$  из подмножества  $\tilde{X}_r$ .
- Параметры  $l$  Гауссовых  $l$ -мерных распределений  $p_r(x_{T_\Delta})$  оцениваются из предположения, что каждое из множеств  $F_r$  является выборкой случайных реализаций  $l$ -мерной случайной величины с гауссовым распределением вероятностей.

Распределение вероятностей  $p_r(x_{T_\Delta})$  следующим образом определяет величины качеств  $q_t(r)$ , используемые в формулировке задачи сегментации (1). Пусть  $x_T : T \rightarrow X$  исходное изображение и  $T_\Delta(t) \subset T$  - часть поля зрения размером  $\Delta \times \Delta$  с центральным пикселем  $t$ . Тогда

$$q_t(r) = \log(p_r(x_{T_\Delta(t)})).$$

Таким образом, задача (1) полностью определена. Ее решение разобьем на два этапа. На первом найдем значение оптимальной разметки  $k_T^*$  в тех пикселях, где это возможно сделать за полиномиальное время, и тем самым упростим задачу. На втором этапе определим значения меток в оставшихся пикселях, пользуясь алгоритмами нахождения приближительного решения [10,17,18].

## 5. Поиск части оптимальной разметки

В случае, когда количество сегментов равно двум ( $L = \{1, 2\}$ ) задача (1) может быть решена точно [10-16].

Если же количество сегментов больше двух ( $L = \{1, 2, \dots, l\}$ ,  $l > 2$ ), то задача является NP-полной ([10]). Тем не менее, можно за полиномиальное время найти часть точного решения, т. е. определить значение оптимальной разметки в некоторых пикселях [8].

Часть точного решения находится из решения некоторых вспомогательных задач. Точное решение вспомогательных задач дает возможность определить значения оптимальной разметки задачи (1) в некоторых областях поля зрения.

Вспомогательная задача строится следующим образом:

- Поле зрения  $T$  и структура  $\mathfrak{Z}$  те же, что и в исходной задаче.
- Фиксируем произвольную метку  $s \in L$ . Множество меток  $L_s$  вспомогательной задачи состоит только из двух меток  $s$  и  $\bar{s}$ .
- Вспомогательная задача состоит в нахождении оптимальной разметки  $k_T^s : T \rightarrow L_s$ :

$$k_T^s = \max_{k_T : T \rightarrow L_s} \left( \sum_{t \in T} q_t^s(k_t) + \sum_{\{t, t'\} \in \mathfrak{Z}} g_{t, t'}(k_t, k_{t'}) \right),$$

где

$$q_t^s(s) = q_t(s), \quad q_t^s(\bar{s}) = \max_{r \neq s} q_t(r)$$

и  $g_{t, t'}(r, r')$  определяется формулой (3).

Поскольку множество меток вспомогательной задачи состоит только из двух элементов, она может быть решена за полиномиальное время (см. [10-16]).

Справедливо утверждение (см. [8]):

Если в некотором пикселе  $t$  значения оптимальной разметки вспомогательной задачи  $k_t^s$  равно  $s$ , то в этом же пикселе значение оптимальной разметки исходной задачи  $k_t^*$  также равно  $s$ .

**Алгоритм**, построенный на этом утверждении, заключается в выполнении для каждого  $s \in L$  следующих операций:

- Построение вспомогательной задачи для метки  $s$ ;
- Поиск оптимальной разметки  $k_T^s$  вспомогательной задачи;
- Если значение  $k_t^s$  в пикселе  $t$  равно  $s$ , то оптимальная разметка  $k_t^*$  исходной задачи (1) также принимает значение  $s$  в пикселе  $t$  ( $k_t^* = s$ ).

## 6. Поиск приближительного решения

Алгоритм, описанный в предыдущем разделе, определяет значения оптимальной разметки в некоторых областях поля зрения. Обозначим через  $T'$  множество пикселей, в которых значение оптимальной разметки осталось неизвестным после нахождения части оптимальной разметки. Задача (1) может быть переписана следующим образом.

$$k_{T'}^* = \arg \max_{k_{T'} : T' \rightarrow L} \left( \sum_{t \in T'} q'_t(k_t) + \sum_{\{t, t'\} \in \mathfrak{Z}'} g_{t, t'}(k_t, k_{t'}) \right), \quad (6)$$

где

$$q'_t(k_t) = q_t(k_t) + \sum_{\substack{t' \in T' \\ \{t, t'\} \in \mathfrak{Z}}} g_{t, t'}(k_t, k_{t'})$$

и

$$\mathfrak{Z}' = \mathfrak{Z} \cap (T' \times T').$$

Новая задача (6) имеет тот же самый вид, что и (1). В действительности же, задача (6) зачастую вычислительно намного проще задачи (1). В самом деле, она часто может быть разложена на независимые подзадачи поиска разметки, которые соответствуют разложениям множества

$T' = \bigcup_{i=1}^n T_i$  на непересекающиеся подмножества, никакие два из которых не являются соседними

$(\forall i \neq j: T_i \cap T_j = \emptyset; \nexists \{t, t'\} \in \mathfrak{Z}: t \in T_i, t' \in T_j, i \neq j)$ .

Для решения каждой подзадачи необходимо использовать соответствующий метод в зависимости от сложности структуры подзадачи.

- Если структура подзадачи проста, например, дерево, то оптимальное решение может быть найдено при помощи метода динамического программирования и его обобщений.
- Если структура подзадачи не позволяет найти точное решение за приемлемое время, то необходимо использовать алгоритмы нахождения приближительного решения. Например, алгоритм моделируемого отжига (simulated annealing) (см. ([17,18]) или же более соответствующий данному случаю " $\alpha$  - expansion" алгоритм, предложенный в [10].

## 7. Экспериментальная проверка алгоритма

Следующие два эксперимента демонстрируют предложенный подход. В обоих примерах параметры марковского случайного поля для каждой из текстур оценены при помощи алгоритмов, описанных в [2-6].

В первом примере изображение (рис. 1(b)) с тремя текстурами было искусственно сгенерировано (см. [7]). Благодаря этому действительная сегментация (рис. 1(a)) известна. Вторым пример (рис. 2) показывает сегментацию аэрофотоснимка (рис. 2(a)). На рисунках 1(c) и 2(b) показаны результаты поиска части оптимальной разметки. Черным цветом обозначены те пиксели, в которых не было найдено решения. Для поиска решения в этих пикселах был применен " $\alpha$  - expansion" алгоритм поиска приближительного решения, описанный в [10]. Результирующие сегментации представлены на рисунках 1(d) и 2(c).

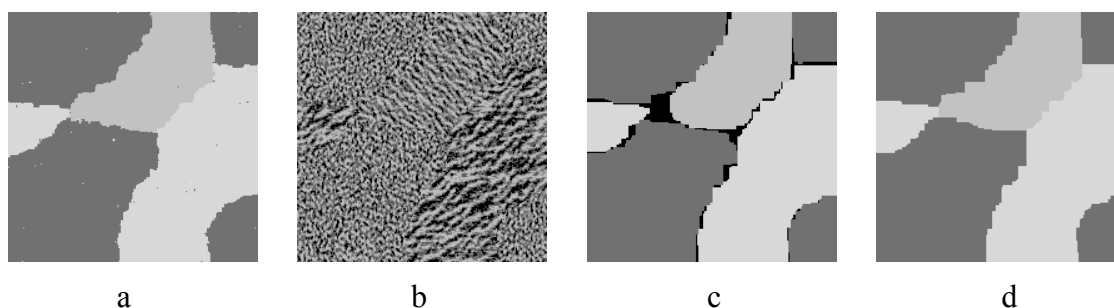


Рис. 1. Сегментация искусственного изображения: а - действительная сегментация; б - сгенерированное искусственное изображение; с - найдено больше, чем 95% оптимальной разметки; d - приближительное решение.

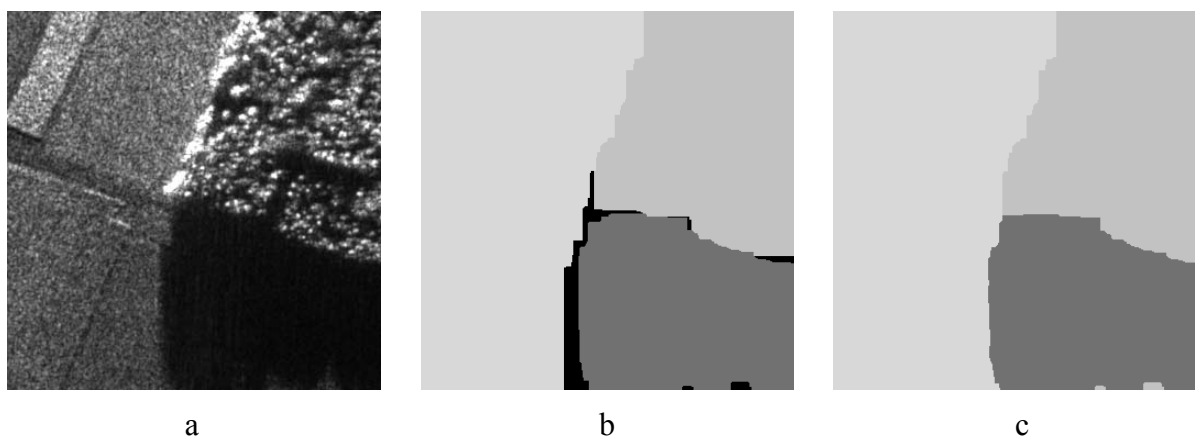


Рис. 2. Сегментация аэрофотоснимка: а - исходное изображение; б - найдено более чем 95% оптимальной разметки; с - приближительное решение.

## 8. Заключение

Предложенный новый алгоритм текстурной сегментации изображений позволяет получить быстрые и надежные результаты, что достигается использованием нового метода подсчета локальных качеств на основании марковских случайных полей общего вида. Алгоритм объединяет метод моделирования текстур марковскими случайными полями и последние достижения в решении (max,+) задач.

1. *Stan Z. Li.* Markov Random Field Modeling in Image Analysis // *Comp. Sci. Workbench*, Springer, 2001. - 323 p.
2. *Gimel'farb G. L.* Texture Modelling by Multiple Pairwise Pixel Interactions // *IEEE Trans.on PAMI.* - 1996. - 18:11. - P. 1110-1114.
3. *Gimel'farb G. L.* Modeling image textures with Gibbs random fields // *Patt. Recog. Lett.* - 1999. - 20:12. - P. 1123 - 1132.
4. *Gimel'farb G. L.* Image Textures and Gibbs Random Fields. - Dordrecht: Kluwer Acad. Publ., 1999. - 250 p.
5. *Zalesny A.* Analysis and Synthesis of Textures with Pairwise Signal Interactions // *Techn. Report: Project 50542 (EC ACTS 0740-"Vanguard", 1.3.1998-1.3.1999).*
6. *Zalesny A., Luc Van Gool.* A Compact Model for Viewpoint Dependent Texture Synthesis // *SMILE 2000 Workshop, LNCS.* - 2001. - **2018**, - P. 124-143.
7. *Ковтун И.* Текстурная сегментация изображений на основании марковских случайных полей // *УСиМ*, - 2003. N 4. - P. 46-55.
8. *Ковтун И.* Поиск части оптимальной разметки некоторого NP-полного подкласса (max,+) задач // *УСиМ*. - 2003. N 6. - P. 33-38.
9. *Kolmogorov V., Zabih R.* What energy functions can be minimized via graph cuts // *ECCV.* - 2002. - **2352** in LNCS, - P. 65-81.
10. *Boykov Y., Veksler O., Zabih R.* Fast approximate energy minimization via graph cuts // *IEEE Trans. on PAMI.* - 2001. - **23**, N 11. - P. 1222-1239.
11. *Ishikawa H., Geiger D.* Segmentation by grouping junctions // *IEEE Comp. Soci. Conf. on Comp. Vision and Patt. Recog.* - 1998. - P. 125-131.
12. *Шлезингер М. И.* Синтаксический анализ двумерных зрительных сигналов в условиях помех // *Кибернетика.* - 1976. - N 4. - С. 113-130.
13. *Шлезингер М. И., Коваль В. К.* Двумерное программирование в задачах анализа изображений // *Автоматика и телемеханика.* - 1976. - N 2. - С. 149-168.
14. *Schlesinger M. I., Flach B.* Some solvable subclass of structural recognition problems // *Czech Patt. Recog. Workshop 2000, - Praha, February 2000.* - P. 55-61.
15. *Schlesinger M. I., Flach B.* Analyzis of optimal labelling problems and their applications to image segmentation and binocular stereovision // *Proc. East-West-Vision 2002 (EWV'02), Intern. Workshop & Project Festival on Comp. Vision, Comp. Graphics, New Media.* - P. 55-60.
16. *Greig D. M., Porteous B. T., Seheult A. H.* Exact Maximum a Posteriori Estimation for Binary Images // *J. Royal Statist. Soc.* - 1989. - Ser. B. - **51**, N 2. - P. 271-279.
17. *Kirkpatrick S., Gellatt C. D., Vecch Jr. et al.* Optimization by simulated annealing // *Science.* - 1983. - **220**, N 4598. - P. 671-680.
18. *Geman S., Geman D.* Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images // *IEEE Trans. on PAMI.* - 1984. - **6**, N 6. - P. 721-741.