

Отчёт по практическому заданию №2. Алгоритмы минимизации энергии для задачи склеивания панорам.

Ю. Н. Лукашкина

`julialukashkina@gmail.ru`

МГУ имени М. В. Ломоносова, Москва

5 апреля 2015 г.

Аннотация

Данный документ содержит отчет по заданию ГМ группы 417 "Алгоритмы минимизации энергии для задачи склеивания панорам".

Содержание

1 Введение	1
2 MRF для склеивания изображений	1
3 Построение графа	2
4 Репараметризация	2
5 α-расширение	2
5.1 Выбор потенциалов	3
6 Эксперименты	3
6.1 Модельные данные	3
6.2 Склеивание двух изображений	3
6.3 Склеивание четырех изображений	5
7 Сравнение с алгоритмом $\alpha\beta$-алгоритмом	5
8 Off-topic	7
9 Литература	7

1 Введение

Марковское случайное поле (MRF) — графическая модель, энергия которой записывается в виде:

$$E(X) = \sum_{i \in \mathcal{V}} \theta_i(y_i) + \sum_{(i,j) \in \mathcal{E}} \theta_{ij}(y_i, y_j), \quad y_i \in \mathcal{P}, \quad (1)$$

где \mathcal{V} — множество индексов переменных, \mathcal{E} — система соседства, $\theta_i : \mathcal{P} \rightarrow \mathbb{R}$ — унарные потенциалы, $\theta_{ij} : \mathcal{P} \times \mathcal{P} \rightarrow \mathbb{R}$ — парные потенциалы. В сумме по рёбрам $(i, j) \in \mathcal{E}$ каждое ребро графа учитывается только один раз. Рассмотрим модель со следующими ограничениями:

- переменные y_p дискретны и принимают значения из множества $\mathcal{P} = \{1, \dots, K\}$, $K \geq 2$,
- система соседства \mathcal{E} — прямоугольная решетка.

Будем реализовывать следующий алгоритм минимизации энергии: алгоритм α -расширение.

2 MRF для склеивания изображений

Задача склеивания изображений состоит в построении одного составного изображения на основе набора исходных изображений. В рамках данного задания предполагается, что все исходные изображения выровнены друг относительно друга. В этих условиях задачу можно решать при помощи минимизации энергии, переменные которой соответствуют номеру изображения, из которого взят конкретный пиксель. Для задачи склеивания энергия строится следующим образом:

- Переменные x_p соответствуют пикселям финального изображения.
- Значение каждой переменной соответствует номеру изображения исходного набора, из которого взят цвет соответствующего пикселя.
- Используется стандартная 4-х связная система соседства.
- Унарные потенциалы должны показывать, из каких изображений должны быть взяты некоторые пиксели (так называемые семена).
- Парные потенциалы должны поощрять 1) короткие разрезы и 2) расположение разреза там, где изображения хорошо соответствуют друг другу.

3 Построение графа

Рассмотрим энергию 1, в которой потенциалы удовлетворяют следующим ограничениям:

- унарные потенциалы неотрицательны: $\forall i \in V \quad \theta_i(0) \geq 0, \theta_i(1) \geq 0$;
- парные потенциалы неотрицательны и равны 0 при равенстве связанных переменных:
 $\forall i, j \in \mathcal{E} \quad \theta_{ij}(0, 0) = \theta_{ij}(1, 1) = 0; \theta_{ij}(0, 1) \geq 0, \theta_{ij}(1, 0) \geq 0$.

В этом случае энергию 1 можно записать в следующем виде:

$$E(x) = \sum_{i \in V} (x_i \theta_i(1) + (1 - x_i) \theta_i(0)) + \sum_{\{i, j\} \in \mathcal{E}} (x_i(1 - x_j) \theta_{ij}(1, 0) + x_j(1 - x_i) \theta_{ij}(0, 1)) \quad (2)$$

По энергии 3 построим ориентированный граф $\tilde{G} = (\tilde{V}, \tilde{E})$, разрез которого будет соответствовать присвоению значений переменным x :

- Множество вершин $\tilde{V} = V \cup s, t$, где s и t — две дополнительные вершины.
- Множество дуг \tilde{E} строится следующим образом: каждому ребру $\{i, j\} \in \mathcal{E}$ ставим в соответствие две дуги (i, j) и (j, i) (нетерминальные дуги), каждой вершине $i \in V$ ставим в соответствие дуги (s, i) и (i, t) (терминальные дуги).
- Будем считать, что множество истока \mathcal{S} соответствует значению переменных 0, множество стока $\mathcal{T} \sim 1$.
- Веса терминальных дуг: $c(s, i) = \theta_i(1), c(i, t) = \theta_i(0)$, где $i \in V$.
- Веса нетерминальных дуг: $c(i, j) = \theta_{ij}(0, 1), c(j, i) = \theta_{ij}(1, 0)$, где $(i, j) \in \mathcal{E}$, и, по договоренности, $i < j$.
- Любой st -разрез такого графа соответствует разбиению множества \tilde{V} на множества истока \mathcal{S} и стока \mathcal{T} . Присваивание значений переменных строится следующим образом: $i \in \mathcal{S} \rightarrow x_i = 0, i \in \mathcal{T} \rightarrow x_i = 1$. Величина разреза совпадает со значением энергии 3 при таких значениях переменных x . Таким образом, задача минимизации энергии вида 3 эквивалента задаче поиска минимального разреза в графе G .

4 Репараметризация

Рассмотрим, что можно сделать при помощи репараметризаций парных потенциалов. Пусть потенциал ребра $\{i, j\} \in \mathcal{E}$ принимает следующие значения: $\theta_{ij}(0, 0) = a, \theta_{ij}(1, 1) = b, \theta_{ij}(0, 1) = c, \theta_{ij}(1, 0) = d$. Применим следующую последовательность преобразований:

1. $\theta_{ij}(0, 0) := \theta_{ij}(0, 0) - a, \quad \theta_{ij}(0, 1) := \theta_{ij}(0, 1) - a, \quad \theta_i(0) := \theta_i(0) + a;$
2. $\theta_{ij}(0, 1) := \theta_{ij}(0, 1) - c + a, \quad \theta_{ij}(1, 1) := \theta_{ij}(1, 1) - c + a, \quad \theta_j(1) := \theta_j(1) + c - a;$
3. $\theta_{ij}(1, 1) := \theta_{ij}(1, 1) - b + c - a, \quad \theta_{ij}(1, 0) := \theta_{ij}(1, 0) - b + c - a, \quad \theta_i(1) := \theta_i(1) + b - c + a;$

Можно убедиться, что данные преобразования приводят к следующим значениям парного потенциала $\{i, j\}$:
 $\theta_{ij}(0, 0) = \theta_{ij}(1, 1) = \theta_{ij}(0, 1) = 0, \theta_{ij}(1, 0) = d + c - a - b$.

Если после этого сделать все унарные потенциалы неотрицательными:

$$\theta_i(0) := \theta_i(0) - \delta, \quad \theta_i(1) := \theta_i(1) - \delta. \text{ Здесь } i \in V, \quad \delta = \min(\theta_i(0), \theta_i(1)) \rightarrow \theta_i(0) \geq 0, \theta_i(1) \geq 0,$$

то энергия будет иметь вид 3.

5 α -расширение

Алгоритм α -расширение минимизирует энергию 1 при помощи выполнения шагов между разметками y , каждый из которых гарантированно не увеличивает значение энергии. Неформально каждый шаг позволяет каждой переменной из y либо присвоить выбранное значение α , либо оставить текущее значение. Пусть есть текущая

разметка y^0 и выбрана «расширяемая» метка $\alpha \in K$. Обозначим новую K -значную разметку y^* . Будем строить энергию бинарных переменных $E(x)$ по энергии 1.

- Значение бинарной переменной $x_i = 0$ соответствует $y_i^* = y_i^0$ (значение переменной y_i не меняется); $x_i = 1$ соответствует $y_i^* = \alpha$.
- Унарные потенциалы: $\theta_i(0) = \psi_i(y_i^0)$, $\theta_i(1) = \psi_i(\alpha)$.
- Парные потенциалы: $\theta_{ij}(0,0) = \psi_{ij}(y_i^0, y_j^0)$, $\theta_{ij}(1,1) = \psi_{ij}(\alpha, \alpha)$, $\theta_{ij}(0,1) = \psi_{ij}(y_i^0, \alpha)$, $\theta_{ij}(1,0) = \psi_{ij}(\alpha, y_j^0)$.

5.1 Выбор потенциалов

Унарный потенциал зададим с помощью масок(seeds): будем штрафовать, если в текущем изображении берется не данная маска, а маски других изображений.

Для уменьшения числа параметров, парные потенциалы можно параметризовать следующим образом: $\theta_{ij}(x_i, x_j) = c_{ij}d(x_i, x_j)$.

Выберем метрику расстояния следующим образом: $d(x_i, x_j) = [x_i \neq x_j]$, то есть метрика равна 0, если метки пикселей совпадают, иначе 1.

Коэффициент c_{ij} задается как разница максимального и минимального значений оттенков серого для всех изображений. Вот, например, что получается для изображения Фунтика 5.1, используемого для тестирования склейки 2х изображений.

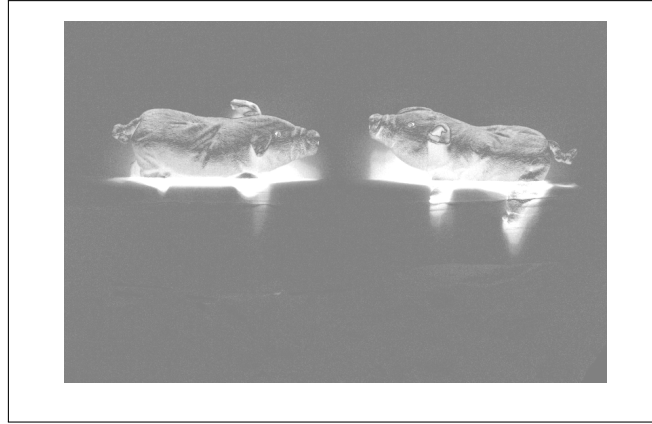


Рис. 1. 50 оттенков Фунтика.

6 Эксперименты

6.1 Модельные данные

Протестируем алгоритм на модельных данных. См. рис. 2. Алгоритм работает.

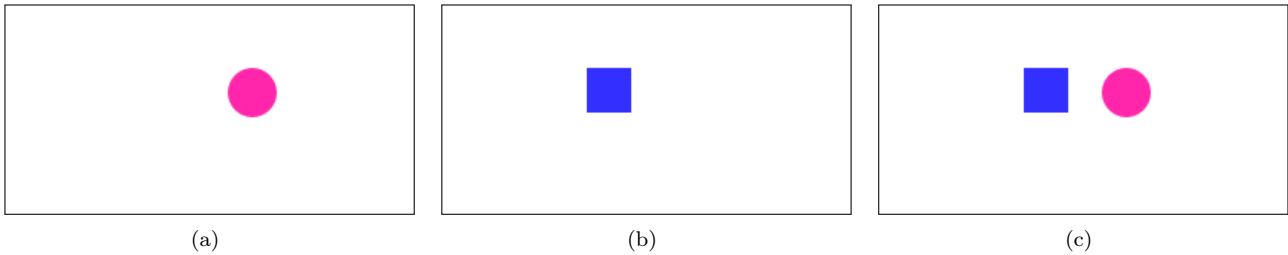


Рис. 2. Пример работы алгоритма на модельных данных.

6.2 Склеивание двух изображений

Теперь рассмотрим задачу склеивания двух изображений. См. рис. 3.



(a)



(b)

Рис. 3. 2 изображения для склейки.

Визуальные дефекты не наблюдаются.



Рис. 4. Склеенное изображение.

6.3 Склеивание четырех изображений

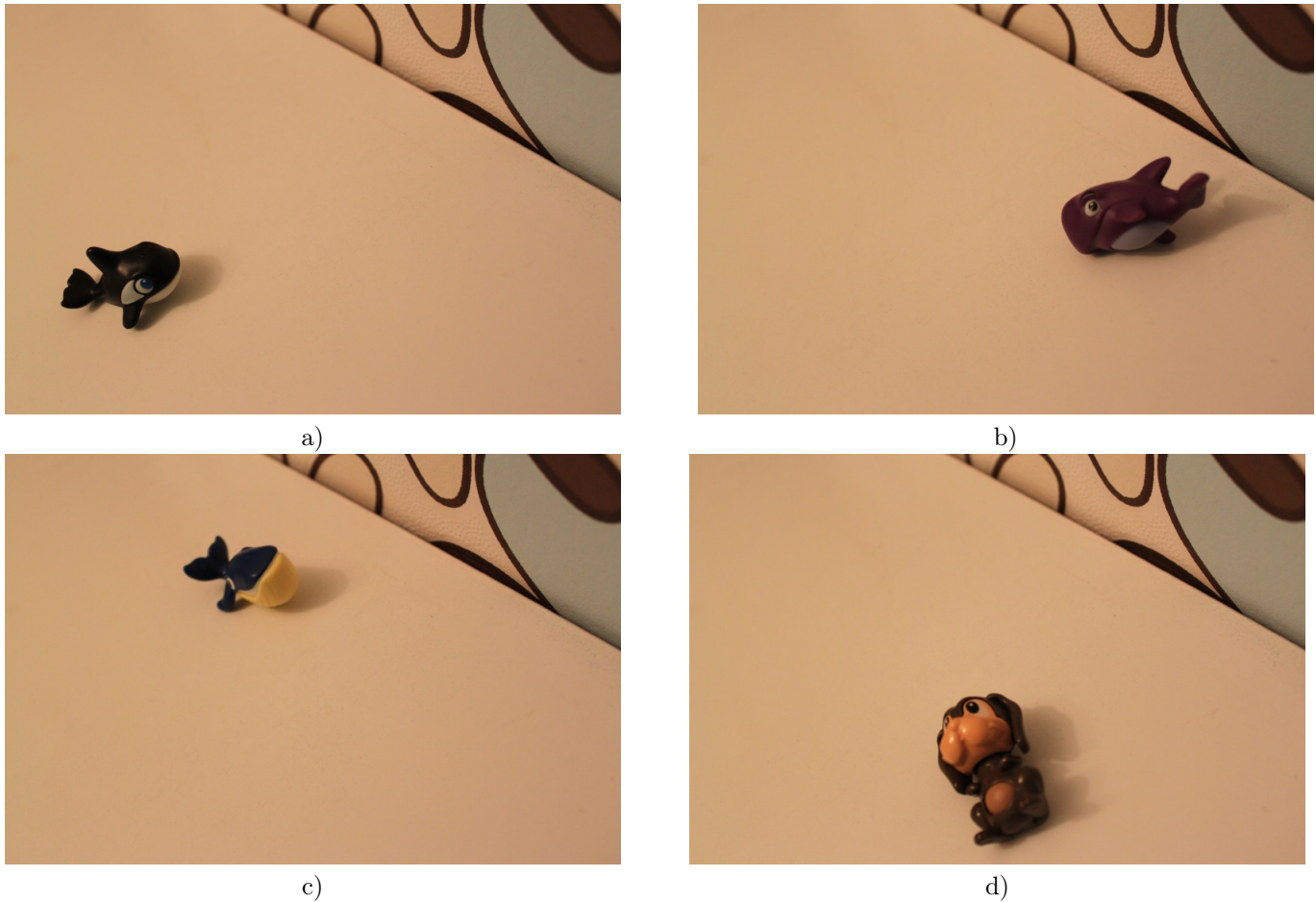


Рис. 5. Изображения для склейки.



Рис. 6. Склеенное изображение.

7 Сравнение с алгоритмом $\alpha\beta$ -алгоритмом

Реализация $\alpha\beta$ -алгоритма была взята у Тюрина Александра. Сравним качество работы алгоритмов для задачи склеивания двух изображений. Оба алгоритма сошлись к одному и тому же значению энергии (821). Алгоритм α расширения за 2 итерации, $\alpha\beta$ -алгоритм за одну. Алгоритм α расширения работал 12.9776 с, $\alpha\beta$ -алгоритма —

1.2460 с. Визуальной разницы не видно 7.



Рис. 7. Склеенное изображение, $\alpha\beta$ -алгоритм.

Сравним качество работы алгоритмов для задачи склеивания 4 изображений. Количество итераций и время работы алгоритмов приведены в таблицах 7, 7. $\alpha\beta$ -алгоритм работает дольше и итоговое значение энергии получается больше, чем итоговое значение α расширения. Также заметны визуальные дефекты на рис.8(b). У Александра его алгоритм $\alpha\beta$ -замены работает хорошо.

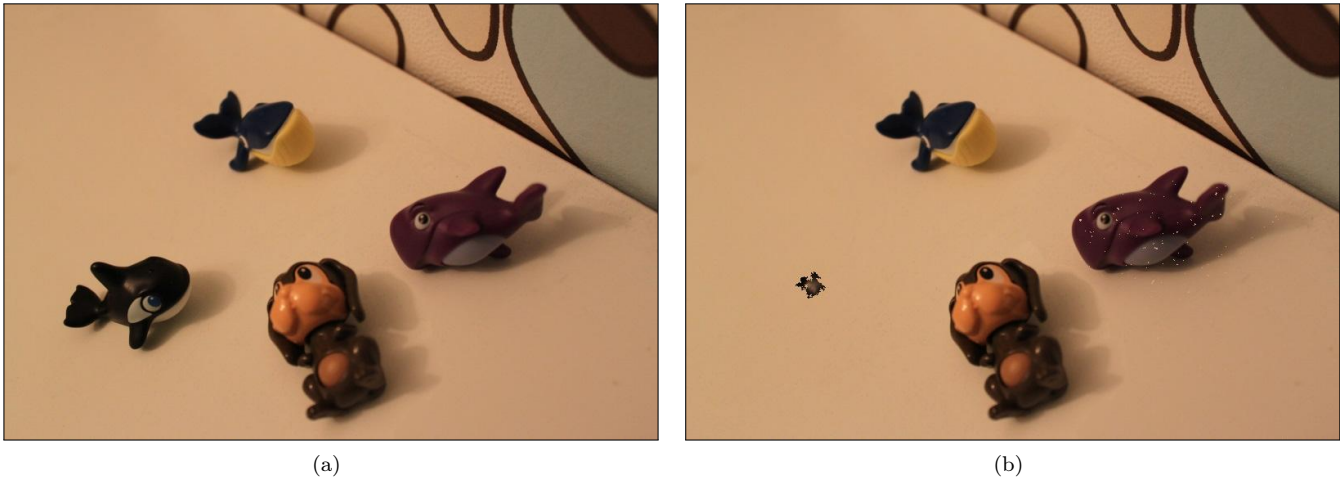


Рис. 8. Склеенное изображение: 8(a) алгоритм α -расширения, 8(b) $\alpha\beta$ -замены.

	итерация 1	итерация 2	итерация 3
значение энергии	5048	4895	4895
время работы (с)	14.0	23.39	32.64

Таблица 1. α -расширение

	итерация 1	итерация 2
значение энергии	114200	114200
время работы (с)	35.0103	67.1031

Таблица 2. $\alpha\beta$ -замена

Все изображения находятся в папке «pics».

8 Off-topic

Мы с коллегой (Муратом Апишевым) решили немного выйти за рамки задания и сделали дополнительный коллаж из изображений двух людей. И вот результат **9!** ¹



(a)



(b)

Рис. 9. Коллаж из двух изображений.

9 Литература

[1] Лекции по ГМ http://www.machinelearning.ru/wiki/images/9/9e/GM_graphCuts.pdf

¹Мы (Юлия Лукашкина и Мурат Апишев) убедительно просим не распространять данные изображения и отчёт в целом за пределами множества лиц, занимающихся проверкой данного задания.