Иркутский государсвенный технический университет

На правах рукописи
УДК xxx.xxx

КАТАШЕВЦЕВ МИХАИЛ ДМИТРИЕВИЧ

АНАЛИЗ КОНТУРНЫХ ИЗОБРАЖЕНИЙ

Специальность 05.13.18 —
«Математическое моделирование, численные методы и комплексы программ»

Диссертация на соискание учёной степени
кандидата физико-математических наук

Научный руководитель:
д. ф-м. н., профессор
Мартьянов В.И.

Иркутск – 2014

Содержание

Глава 1

Контурные изображения

1.1 Базовые понятия

Определение 1.1.1. Растровое изображение есть функция

$$I_{rgb}(x, y) : N \times N \to ([0, 255], [0, 255], [0, 255])$$

Таким образом каждой точке (x, y) мы сопоставляем тройку (r, g, b). Первый элемент тройки соответствует красной компоненте цвета в растровом изображении, второй – зеленой и третий – синей. Далее, для краткости, будем использовать следующую запись:

$$\begin{gathered} I_r(x, y) = I_{rgb}(x, y)_r \text{ – красная компонента} \\ I_g(x, y) = I_{rgb}(x, y)_g \text{ – зеленая компонента} \\ I_b(x, y) = I_{rgb}(x, y)_b \text{ – синяя компонента} \end{gathered}$$

Определение 1.1.2. Определим растровое изображение заданного в оттенках серого как функцию

$$I_{grey}(x, y) : N \times N \to [0, 255]$$

Определение 1.1.3. Введем оператор «обесцвечивания» D, который обеспечивает переход от цветного изображения I_{rgb} к изображению I_{grey}, заданному в оттенках серого:

$$D(I_{rgb}) = I_{grey}$$

Существует несколько основных способов обесцвечивания изображения:

1. Красный канал $D_{red}(I_{rgb}) = I_r$
2. Зеленый канал $D_{greeb}(I_{rgb}) = I_g$
3. Синий канал $D_{blue}(I_{rgb}) = I_b$
4. Среднее значение (average): $D_{avg}(I_{rgb}) = \frac{I_r + I_g + I_b}{3}$
5. Лума (luma), учитывает особенности восприятия цвета человеком:

   $$D_{luma}(I_rgb) = I_r \cdot 0.3 + I_g \cdot 0.59 + I_b \cdot 0.11$$

   Значения коэффициентов, иногда, могут отличаться от приведенных выше, но их сумма всегда равна 1
6. Минимум $D_{min}(I_{rgb}) = min(I_r, I_g, I_b)$

7. Максимум $D_{min}(I_{rgb}) = max(I_r, I_g, I_b)$

8. Обесцвечивание (desaturtaion): $D_{desaturation}(I_{rgb}) = \frac{D_{min}(I_{rgb})+D_{max}(I_{rgb})}{2}$

Замечание 1.1. Наилучший результат для средне-статистического изображения (с гистограммой близкой к нормальной) получается при использование 5-го и последнего способов. Под наилучшим результатом понимается сохранение яркости (компоненты value в модели HSV) цветов исходного изображения.

Определение 1.1.4. Растровое монохромное изображение есть функция

$$I_m(x, y) : N \times N \to \{0, 1\}$$

Переход от изображения заданного в оттенках серого к монохромному изображению осуществляется через операцию отсечения. Операция отсечения реализуется через оператор отсечения T, для некоторого фиксированного $t \in [0, 255]$

Определение 1.1.5. Оператор отсечения T есть:

$$T(t, I_{grey}) = \left\{ \begin{array}{ll} 1 & , I_{grey} < t \\ 0 & , \text{иначе} \end{array} \right\}$$

Таким образом монохромное изображение есть $I_m = T(t, I_{grey})$

Замечание 1.2. Значение t определяется опытном путем и зависит от исходного изображения. Для рукописного текста написанного черной ручкой на офисной бумаге берутся значения близкие к 100 (чем меньше значение, тем темнее изображение).

Замечание 1.3. В данной работе не рассматриваются методы адаптивного отсечения, в силу их медленной производительности излишней в данном контексте точности.

Определение 1.1.6. Точки (x, y) для которых верно $I(x, y) = 1$ будем называть заполненными.

Определение 1.1.7. Точки (x, y) для которых верно $I(x, y) = 0$ будем называть пустыми.

Множество заполненных точек образует множество связных областей

$$\{K_1, K_2 \dots K_n\},$$

таких что:

1. $\forall(x_1, y_1)\forall(x_2, y_2)(|x_1 - x_2| > 1 \wedge |y_1 - y_2| > 1)$, где $(x_1, y_1) \in K_i$, $(x_2, y_2) \in K_j$ и $i \neq j$

2. $\forall(x_1, y_1)\exists(x_2, y_2)(|x_1-x_2| \leq 1 \wedge |y_1-y_2| \leq 1)$, где $(x_1, y_1), (x_2, y_2) \in K_i$ и $(x_1, y_1) \neq (x_2, y_2)$

3. $|K_i| \geq 2$

Определение 1.1.8. Всякую связную область K_i будем называть контуром

Определение 1.1.9. Растровое изображение I содержащие по крайней мере один контур будем называть растровым контурным изображением

Замечание 1.4. Не исключая общности, далее будут рассматриваться только растровые изображения содержащие один контур, а под растровым контурным изображением будет пониматься растровое изображение содержащие только один контур.

1.2 Модель описания растровых контурных изображений

В качестве математической модели представления растрового контурного изображения будем использовать четырех-основную алгебраическую систему вида.

Определение 1.2.1. Контурное изображение (далее, изображение) есть система вида

$$\mathfrak{M} =< A, R, V, M; Sector, Angle, Metric, Relation > \tag{1.1}$$

где

A – множество всевозможных дуг,

R – множество связей дуг,

$V \subset Z$ – множество допустимых углов (например от 0 до 360 градусов),

$M \subset Z$ – множество относительных мер,

$Sector : A \to V$ – задает градусную меру дуги,

$Metric : A \to M$ – функция сопоставляющая каждой дуге ее относительную величину,

$Angle : R \to V$ – задает угол соединения двух дуг

$Relation : R \to A \times A$ – сопоставляет каждой связи дуги, те дуги, которые она соединяет.

Замечание 1.5. Важно отметить, что в этом представлении все множества являются конечными, и, если множества A и R являются фиктивными (чисто техническими элементами) данной модели, и определяются через функции, то множество $V = \{v_0, v_1, ..., v_n\}$ – есть конечное множество чисел, с максимальным v_{max} и минимальным v_{min} элементами, разбитое шагом δ на $n + 1$ элементов, где

$$n = \frac{v_{max} - v_{min}}{\delta}$$
$$v_0 = v_{min}$$
$$v_i - v_{i-1} = \delta, \forall i = \bar{1, n}$$
$$v_n = v_{max}$$

Замечание 1.6. Как и множество V, множество M конечно. Однако выбор верхней границы для M не столь очевиден, так как не исключена возможность того, что разница в размерах между двумя дугами может быть весьма существенна (например в несколько миллионов раз). Однако, в рамках нашей области применения (распознавание символов), когда в качестве «эталонного наблюдателя» выступает человеческий глаз, разница что в 1000, что в 1000000 раз почти неразличима, и поэтому ею вполне можно пренебречь, выбрав в качестве максимального значения например 100000 процентов, а в качестве шага одну десятую процента. Таким образом всякая дуга может быть как больше так и меньше любой дуги не более чем в 1000 раз.

Для наших целей важно всегда работать только с конечными множествами, что достигается рассмотрением конечных множеств A, R, а также предположением о наличии минимального шага возрастания количественных характеристик дуг и связей дуг.

Таким образом контурное изображение будет представляет собой систему дуг и связей дуг. Где всякая дуга определяется через ее градусную меру и через относительную (в данном контуре) длину дуги. А всякая связь определяется через угол связи и пару дуг которые она связывает.

1.3 Преобразование растрового изображения

Переход от растрового контурного изображения к изображению состоит из двух этапов. Первый этап — волновая скелетизация. С помощью скелетезации на основе растрового изображение строится граф (скелет), который визуально адекватно соответствует исходному изображению.

Пусть I – растровое контурное изображение и K – есть его контур.

Определение 1.3.1. Точку $q(x_1, y_1)$ будем называть соседом точки $p(x, y)$ если $|x - x_1| \leq 1$ и $|y - y_1| \leq 1$ и $p \neq q$. Введем отношение соседства $N(p, q)$, которое истинно если p сосед q.

Очевидно что точка p не может иметь более 8 соседей. Обозначим через N_K^p множество всех соседей точки $p(x, y)$ лежащих контуре K:

$$N_K^p = \{q | q \in K \wedge N(p, q)\}$$

Согласно определению контура (1.1.8) очевидно, что K не имеет изолированных точек т.е.

$$\forall p \exists q : N(p, q)$$

$$p, q \in K$$

Замечание 1.7. Скелетом I будем называть граф $G(V, E)$, «интуитивно адекватно отражающий» исходное изображение.

Определение 1.3.2. Волной w будем называть конечное множество точек $\{p_j\}$.

Определение 1.3.3. Множество волн $\{w_1, w_2, ..., w_n\}$ будем называть подволнами волны w если:

$$\bigcup_{i=1,n} w_i = w$$

$$w_i \cap w_j = \emptyset,\ i, j = \overline{1, n},\ i \neq j;$$

и для любых двух точек $p \in w_i,\ g \in w_j$, где $i \neq j$ верно $\neg N(p, g)$.

Введем функцию вычисляющую центр масс точек волны

$$g(w) = \frac{\sum\limits_{p \in w} p}{|w|}$$

1.3.1 Волновая скелетизация

Опишем алгоритм построения скелета растрового контурного изображения.

Зададим начальные условия. В качестве начальной волны подойдет любая точка из F. Имеем следующую начальную конфигурацию:
${w_0}^0 = \{p\}, p \in K$ – начальная волна,
$W_0 = \{{w_0}^0\}$ – множество волн,
$F_0 = K$ – состояние заполненной области,
$G_0(V_0, E_0), V_0 = \{p\}, E_0 = \emptyset$ - начальное состояние скелета.

Определим n-ый шаг итерации следующим образом. Для всякой i-ой волны ${w_i}^{k_i - 1}$ из W_{n-1} (k_i - соответствует k_i-ой итерации w_i):

$$ {w_i}^{k_i} = \bigcup_{p \in {w_i}^{k_i - 1}} N_{F_{n-1}}^p \setminus \bigcup_{j<i} {w_j}^{k_j - 1} \tag{1.2}$$

Если $u_1, \ldots, u_m$ есть подволны волны ${w_i}^{k_i}$, тогда

$$W_n^i = \{w_{l+1}, \ldots, w_{l+m}\}$$

где

$$w_{l+j} = u_j, j = \overline{1, m}$$

$$l = |W_{n-1}| + \sum_{j<i} |W_n^j|.$$

Ребра в графе образуют вектора, связывающие центры масс полученных подволн с центром массы ${w_i}^{k_i-1}$

$$E_n^i = \{(v_i^{k_{i-1}}, v_{l+j}^{k_i})\}, j = \overline{1, m}$$

$$V_n^i = \{v_{l+j}^{k_i}\}, j = \overline{1, m}$$

$$v_i^j = g(w_i^j)$$

Если же волна $w_i^{k_i}$ не имеет разрывов и $w_i^{k_i} \neq \emptyset$, то

$$W_n^i = \{{w_i}^{k_i}\}$$

$$E_n^i = \{(v_i^{k_{i-1}}, v_i^{k_i})\}$$

$$V_n^i = \{v_i^{k_i}\}$$

Если $w_i^{k_i} = \emptyset$

$$W_n^i = \emptyset, E_n^i = \emptyset, V_n^i = \emptyset$$

Таким образом, при $s_n = |W_{n-1}|$:

$$W_n = \{W_n^i\}, i = \overline{1, s_n}$$

$$V_n = V_{n-1} \bigcup_i V_n^i, i = \overline{1, s_n}$$

$$E_n = E_{n-1} \bigcup_i E_n^i, i = \overline{1, s_n}$$

$$F_n = F_{n-1} \setminus P_{n-1},$$

$$P_{n-1} = \{p : p \in w, w \in W_{n-1}\}$$

Если $|F_n| = 0$ то алгоритм прекращает цикл итераций, а граф

$$G = (V_n, E_n)$$

является скелетом исходного изображения f.

Утверждение 1.3.1. Алгоритм волновая скелетизация, остановится на любом контуре мощности m удовлетворяющей условиям алгоритма.

Доказательство. Пусть $m = 1$, тогда согласно алгоритму

$$|F_1| = |F_0 \setminus \{p : p \in w, w \in W_0\}| = |\{p\} \setminus \{p : p \in w_0\}| = |\{p\} \setminus \{p\}| = \emptyset$$

следовательно алгоритм прекращает свою работу а граф $G = (V_1, E_1) = (\{p\}, \emptyset)$ является скелетом изображения.

Пусть $m > 2$ и существует такое l что для всякого $k < l, |F_k| < |F_{k-1}|$ и $|F_l| = |F_{l-1}|$, тогда $P_{n-1} = \emptyset$, отсюда следует, что $\{p : p \in w, w \in W_{n-1}\} = \emptyset$, а это возможно только в двух случаях:

1. если W_{n-1} пусто, тогда, в силу и в силу отсутствия изолированных точек, $F_{n-2} = \emptyset$, получаем противоречие с условием остановки.

2. если $\forall w \in W : |w| = 0$, тогда, опять же в силу и в силу отсутствия изолированных точек, получаем что $F_{n-2} = \emptyset$, снова получаем противоречие с условием остановки.

Следовательно такого l не существует, и алгоритм сходится для всякой непустой заполненной области без изолированных точек. □

1.3.2 Построение модели $\mathfrak{M}$ для графа G

Прежде чем дать формальное описание алгоритма, рассмотрим как он работает неформально:

1. Для каждого простого пути выполняется:
   (a) Разбиение пути по точкам смены направления обхода
   (b) Для каждого разбиения выполнятся
      i. Разбиение спиралей. Чтобы определить закручен ли путь по спирали, надо проверить пересекает ли хорда путь. Если пересечение есть, то необходимо разбить путь точками пересечения
      ii. Для каждого разбиения выполняется:
         A. Разбиение по точкам перегиба. Точками перегиба считаются образующие две дуги отклонившиеся от угла идеального соединения. Угол γ идеального соединения двух дуг градусной меры α и β:

$$\gamma = 2\pi - (\alpha + \beta)$$

2. Результатом п.1 является множество подпутей, каждый из которых переводится в дугу. Градусная мера дуги вычисляется с использование формулы Гюйгенса. Определив наиболее удаленную точку пути от хорды стягивающей путь, и вычислив её положение относительно хорды направленной от начала к концу пути, мы определяем направление обхода. Если точка слева, то обход ведется по часовой стрелке, если точка справа — обход ведется против часовой стрелки, если же точка лежит на прямой, то верны оба утверждения.

3. Расчет связей дуг. Связь между двумя дугами существует, если пути образующие дуги имели общие вершины. Угол соединения между дугами рассчитывается, как угол между стягивающими их хордами

Замечание 1.8. Стоит отметить что разбиение на дуги, как правило, выполняется на интерполированном графе в котором часть узлов удаленно в силу их избыточности.

Замечание 1.9. Хотя разбиение и может быть использовано как есть, на практике полезнее добавлять возможность вариации параметров, например, допускать возможность некоторого отклонения от угла идеального соединения или для точек смены направления расширять область проверки на смену направления обхода.

Пусть $\{P_1, P_2, \ldots, P_m\}$ – множество простых цепей графа $G = (V, E)$ таких что:

1. $P_i = v_1^i, v_2^i, \dots v_{n_i}^i$

2. $\forall v \in P_i \forall u \in P_j (v \neq u)$, где $i \neq j$

3. $d(v_1^i) \neq 2$ и $d(v_{n_i}^i) \neq 2$

4. $\forall j \notin \{1, n_i\} \left[d(v_j^i) = 2\right]$

5. $\bigcup_i P_i = V$

Определение 1.3.4. Пусть $r(p, v)$ – есть растояние от точки p до вектора v со знаком

Определение 1.3.5. Будем говорить что в узле v_i меняется направление обхода простого

$$\{v_1, ..., v_{i-1}, v_i, v_{i+1}, ..., v_n\}$$

в окрестности $\epsilon \in \{2, 3, ...\}$, если

$$Sign(\sum_{j=i-\epsilon} r(v_j, h)) \neq Sign(\sum_{j=i+\epsilon} r(v_j, h))$$

где $h = (v_{i-\epsilon}, v_{i+\epsilon})$

Если соотношение выполняется для $\epsilon = 2$ будем просто говорить, что в точке v_i меняется направление обхода.

Определение 1.3.6. Пусть $P = v_1, \dots, v_n$ – некоторый простой путь, $I = \{i_1, \dots, i_m\}$ – множество индексов $I \in N$, причем $m < n$, $i_j < i_{j+1}$, где $j = \overline{1, m}$. Будем называть множество S_I^P – разбиением пути P индексами I, если:

$$S_I^P = \bigcup_{k=\overline{0,m-1}} \{v_j \mid i_k \leq j \leq i_{k+1}\}$$

Пусть множество точек $\{v_{j_1^i}, \dots v_{j_{k_i}^i}\}$ множество точек смены направления обхода цепи P_i. Определим множество индексов задающих разбиение пути P_i по направлению обхода

$$I_{P_i}^{dev} = \{j_1^i, \dots j_{k_i}^i\}$$

$$j_k^i < j_{k+1}^i$$

Определение 1.3.7. Определим множество индексов узлов задающие разбиение подпути $P = v_1, \dots v_n$ закрученного в спираль следующим образом

$$I_P^{spir} = \left\{ \min_{|r(h,v_i)|} (i, i+1) \mid Sign(r(h, v_i)) \neq Sign(r(h, v_{i+1})) \wedge i = 1, n-1 \right\}$$

$$h = (v_1, v_n)$$

Определение 1.3.8. Определим функцию сопоставляющую каждой паре $k, l \in P$, где $k, l < n$, путь находящийся между ними в простой цепи $P = v_1, \dots, v_n$.

$$f_{path}^P(k, l) \sim v_k, v_{k+1}, \dots, v_l$$

Определение 1.3.9. Разбиением пути P на характеристические подпути будем считать разбиение определенное следующим образом:

$$S_{P_i}^{dev} = \bigcup_{l \in I_{P_i}^{dev}} \{\{v_t \mid t \in \{j \mid i_l \leq j \leq i_{l+1}\}\}\}$$

$$S_{P_i} = \bigcup_{j \in I_P^{spir},\, P \in S_{P_i}^{dev}} \{\{v_t \mid t \in \{j \mid i_l \leq j \leq i_{l+1}\}\}\}$$

Рисунок 1.1: Схема разбиения графа «особыми» точками

Литература

1. Название статьи / Автор1, Автор2, Автор3 [и др.] // Журнал. 2012. Т. 1. С. 100.
2. Автор. Название книги / под ред. Редактор. Издательство, 2012.
3. Автор. название тезисов конференции // Название сборника. 2012.
4. Название буклета.
5. “This is english article” / Author1, Author2, Author3 et al. // Journal. 2012. Vol. 2. P. 200.