

Выделение нечетких объектов на цветных изображениях с помощью стохастических алгоритмов¹

Д. Сидорчук*, Е. Жижина**

*Национальный исследовательский университет “Высшая школа экономики”, Москва, Россия

**Институт проблем передачи информации им. А.А. Харкевича,
Российская академия наук, Москва, Россия

Поступила в редколлегию 15.08.2013

Аннотация—На данный момент проблему выделения лиц небольшого количества людей с близкого расстояния можно считать решенной: это «умеет» делать любой современный цифровой фотоаппарат или мобильный телефон. Наш интерес в данной работе представляют экстремальные случаи. До сих пор открытой задачей является выделение большого числа объектов с нечеткими границами, небольшими размерами, возможным перекрытием друг друга. Эту задачу мы решаем, применяя стохастический алгоритм MBAD (multiple birth and death algorithm), который до этого не использовался для выделения людей на изображениях, но успешно применялся в других задачах компьютерного зрения. Работа посвящена объяснению основных принципов действия алгоритма, а также описанию его модификаций для задачи выделения лиц в плотной толпе. Помимо стохастического алгоритма, мы вводим в этой работе новый детектор для выделения нечетких лиц на изображениях, основанный на той же самой функции энергии, которую мы используем в стохастическом подходе.

Ключевые слова. Выделение лиц в толпе, динамика рождения-гибели в непрерывном пространстве, стохастическое моделирование, процедура аннилинга

1. ВВЕДЕНИЕ

Работа посвящена одной из задач обработки изображений: проблеме выделения объектов на цветных изображениях в автоматическом режиме. Эта проблема уже давно широко исследуется учеными, и во многих частных случаях достигнуты хорошие результаты. Данное исследование посвящено одному из возможных методов решения этой проблемы, основанном на стохастическом MBAD (Multiple birth and death) алгоритме, моделирующем случайный процесс. Кроме стохастического алгоритма, мы вводим в этой работе новый детектор для выделения нечетких лиц, основанный на применении той же самой функции энергии, которую мы используем в стохастическом подходе.

Случайный процесс, который лежит в основе алгоритма – это стохастическая динамика рождения-гибели, которая является одним из примеров стохастических динамик многочастичных систем. Выделить объекты в данном случае означает получить набор точек, соответствующий центрам этих объектов. Для моделирования этой динамики используется метод Монте-Карло.

Стохастические алгоритмы, основанные на применении динамики рождения-гибели известны под названием RJMCMC (Reversible-jump Markov chain Monte Carlo), см., например, [1]. Новый подход, основанный на применении динамик рождения-гибели, был недавно развит в цикле работ Кс. Декомба (ИНРИА, Франция) и Р.А. Минлоса, Е.А. Жижинной (ИППИ РАН,

¹ Работа выполнена при частичной поддержке гранта РФФИ 11-01-00485а.

Россия) [2], [3]. В этих работах было дано теоретическое обоснование алгоритма и продемонстрированы возможности этого метода для разных задач выделения объектов. Разработанный алгоритм получил название MBAD. В дальнейшем алгоритм применялся также для распознавания зданий, см. [4], крон деревьев на аэрокосмических снимках лесных массивов, см. [2], [5], сети дорог, см. [6], стай птиц, см. [2]. Из перечисленного наиболее близкой задачей является задача выделения верхушек деревьев при аэрокосмической съемке, описанная в статье [2], поскольку фотографии, которые использовались для проверки работы алгоритма, по форме, размеру и количеству выделяемых объектов в некоторой степени схожи с нашими данными. Работа [2] в определенном смысле является базовой для данного исследования.

Цель данной работы состоит в исследовании возможности использовать разработанный подход для выделения лиц в плотной толпе людей и применительно к цветным изображениям (до этого обрабатывались только черно-белые). Одним из преимуществ рассматриваемого здесь подхода к проблеме выделения лиц является то, что он работает с изображениями большой толпы людей, когда нет возможности использовать для выделения детали лица (такие как нос, рот, глаза). Заметим, что в отличие от большинства других подходов, этот подход не связан с машинным обучением.

В настоящее время основным методом поиска лиц на изображении (или видео) является алгоритм, представленный в 2001 году Полом Виолой и Майклом Джонсом, см. [7]. Алгоритм этот основан на применении детектора лиц, использующего признаки Хаара. Для работы этого метода нужно, чтобы качество изображения позволяло использовать разницу в интенсивностях цвета пикселей лица: например, зоны около глаз и рта обычно более темные, чем в среднем лице. Существуют работы, посвященные выделению людей в плотной толпе, однако, в этих работах делается упор на небольшие группы людей. Представленные в них алгоритмы успешно обрабатывают изображения толпы размером около 50 человек. Например, в работе [8], посвященной обработке изображений плотной толпы, используется каскадная классификация Виолы–Джонса методом AdaBoost. Авторы добились хороших результатов (средний процент верно выделенных объектов 76.8%), но важно отметить, что среднее количество людей на обрабатываемых видео фрагментах составило 35.35 человека, что на порядок меньше чем в нашей работе. Среди других исследований на схожие темы стоит отметить выделение фигур людей и отдельных объектов сложной формы в работах [9], [10], [11], [12].

Предложенный нами метод изначально нацелен на обработку изображений на порядок большего количества людей, когда изображение отдельного лица – это практически однородное светлое пятно. В данной работе мы рассматриваем случай, когда размер выделяемых на изображении объектов предполагается практически одним и тем же. Фотографии, которые удовлетворяют нашим требованиям, сделаны с большой высоты под небольшим углом, благодаря чему на снимках запечатлено большое количество людей, но эффект перспективы пренебрежимо мал. От этого ограничения можно отказаться, мы обсудим это ниже, в последнем разделе.

Опишем структуру работы. Во втором разделе рассматривается математическая модель, на которой основывается алгоритм. В третьем разделе подробно описывается функция энергии, которая является наиболее важным элементом данной работы. Четвертый раздел посвящен стохастическому алгоритму, некоторым деталям его компьютерной реализации, а также альтернативному подходу, использующему новый детектор, основанный на функции энергии. В последнем разделе описываются результаты работы нашего метода и перспективы дальнейшего развития.

2. ОПИСАНИЕ МОДЕЛИ

В качестве данных мы имеем цифровое цветное изображение, которое можно определить как отображение I некоторой области двумерной решетки $V \subset Z^2$ в трехмерное пространство

цвета:

$$I : V \rightarrow Y, \quad V \subset \mathbb{Z}^2, \quad Y \subset \mathbb{R}^3.$$

При этом каждому пикселю $x \in V$ соответствует трехмерный вектор $\vec{\omega}_x \in \mathbb{R}^3$, определяющий цвет этого пикселя в кодировке HSV. Решетка \mathbb{Z}^2 используется для дискретизации области непрерывного пространства $\hat{V} \subset \mathbb{R}^2$, где будет определено множество конфигураций дисков, или что эквивалентно, конфигураций точек, которые являются центрами дисков фиксированного радиуса.

Будем понимать под конфигурацией любой конечный неупорядоченный набор точек из ограниченной области V :

$$\gamma = \{x_i\} \in \Gamma(V), \quad x_i \in V.$$

Мы выбираем заранее и фиксируем радиус r , который зависит от размера объектов на обрабатываемом изображении и не меняется в ходе обработки данного изображения (или группы подобных изображений).

Наша задача – выделить все требуемые объекты на изображении I и получить конфигурацию γ^* , состоящую из точек, являющихся центрами всех выделенных объектов. Для решения этой задачи используется так называемый гиббсовский подход, основанный на применении гиббсовских случайных полей к задачам обработки изображений. Суть подхода заключается в том, что на пространстве конфигураций $\Gamma(V)$ вводится (апостериорное) распределение вероятностей, заданное в гиббсовской форме:

$$\mu_\beta(d\gamma) = \frac{z^n}{Z_\beta} e^{-\beta H(\gamma)} \lambda(d\gamma), \quad \gamma \in \Gamma(V). \quad (1)$$

Здесь $n = |\gamma|$ – количество точек конфигурации, Z_β – нормирующий множитель, β, z – это параметры: β – так называемая обратная температура, z – параметр интенсивности априорного точечного поля, $H(\gamma)$ – функция энергии конфигурации. Это распределение является так называемой гиббсовской перестройкой меры Лебега–Пуассона $\lambda(d\gamma)$ с интенсивностью 1 и позволяет нам с помощью функции $H(\gamma)$ учитывать информацию о конкретном расположении точек конфигурации. Функция энергии играет ключевую роль в данной работе и позже мы подробно ее обсудим.

Таким образом, выбирая функцию в соответствии с конкретными условиями задачи, мы можем переформулировать задачу отыскания конфигурации γ^* как оптимизационную задачу: решением поставленной проблемы является конфигурация, на которой достигается минимум функции энергии

$$\gamma^* = \arg \max \mu_\beta(\gamma) = \arg \min H(\gamma). \quad (2)$$

Заметим, что конфигурация, определенная в (2), может быть не единственна. В работе [2] доказан уточненный результат о том, что слабый предел последовательности мер μ_β при $\beta \rightarrow \infty$ сходится к мере μ_∞ , которая сосредоточена на конфигурациях (2), содержащих минимально возможное число точек. Однако эта форма описания конфигураций γ^* не является конструктивной, она не дает способа построения таких конфигураций. В работах [2], [13] была построена стохастическая неравновесная динамика, сходящаяся к μ_∞ . Эта динамика представляет собой случайный процесс рождения гибели с однородными интенсивностями рождения z и интенсивностями гибели $e^{-\beta(H(\gamma \setminus \{x\}) - H(\gamma))}$. Легко видеть, что такая динамика при фиксированном β является обратимой:

$$\mu(\gamma) c(\gamma \rightarrow \gamma') = c(\gamma' \rightarrow \gamma) \mu(\gamma'), \quad \gamma' = \gamma \setminus \{x\}, \quad (3)$$

где $c(\gamma \rightarrow \gamma')$ – интенсивность перейти от конфигурации γ к конфигурации γ' , и инвариантной мерой является гиббсовская мера μ_β (1). Это равенство можно воспринимать как требование

того, что конфигурация с большей мерой (то есть более вероятная) должна изменяться с малой вероятностью, а конфигурация с малой мерой (то есть менее вероятная) должна изменяться с большей вероятностью. Далее, изменяя параметр $\beta = \beta(t)$ со временем, мы получаем из описанной выше динамики уже неравновесную динамику. При этом если параметр $\beta(t)$ изменять согласно процедуре аннилинга $\beta(t) \leq c \ln t$, см., например [14], [15], то в результате получится динамика, сходящаяся к μ_∞ , см. [13], [16].

Такая динамика является непрерывной по времени, и для того, чтобы использовать ее для моделирования нам необходимо аппроксимировать ее подходящей марковской цепью. Построение аппроксимационного процесса вместе со всеми доказательствами сходимости можно найти в работах [2], [13]. Здесь мы вкратце приведем необходимые конструкции, чтобы не нарушать целостность этой работы. Итак, в качестве аппроксимирующего процесса $\hat{S}_{\beta, \delta}(k)$ мы рассматриваем марковскую цепь над тем же пространством состояний $\Gamma(V)$. Переход из состояния γ в состояние $\tilde{\gamma}$ происходит в два этапа: за счет удаления точек из конфигурации $\gamma : \gamma \rightarrow \gamma_1$, $\gamma_1 \subseteq \gamma$ (часть гибели), и добавления новых точек γ_2 (часть рождения). Таким образом, за один шаг марковской цепи происходит “многоточечное” изменение текущей конфигурации:

$$\gamma \rightarrow \tilde{\gamma}, \quad \tilde{\gamma} = \gamma_1 \cup \gamma_2, \quad \gamma_1 \subseteq \gamma.$$

Переходные вероятности отдельных событий задаются следующим образом:

– для “гибели” точки x из конфигурации γ :

$$p_{x, \delta_k}^d = \begin{cases} \frac{\alpha_k(x, \gamma) \delta_k}{1 + \alpha_k(x, \gamma) \delta_k}, & \gamma \rightarrow \gamma \setminus x \\ \frac{1}{1 + \alpha_k(x, \gamma) \delta_k}, & \gamma \rightarrow \gamma \quad (\text{т.е. } x \text{ сохраняется}), \end{cases} \quad (4)$$

где

$$\alpha_k(x, \gamma) = e^{-\beta_k(H(\gamma \setminus \{x\}) - H(\gamma))},$$

– для “рождения” новой точки:

$$p_{x, \delta_k}^b = \begin{cases} z \Delta y \delta_k, & \gamma \rightarrow \gamma \cup y \\ 1 - z \Delta y \delta_k, & \gamma \rightarrow \gamma \quad (\text{т.е. новая точка не возникла}). \end{cases} \quad (5)$$

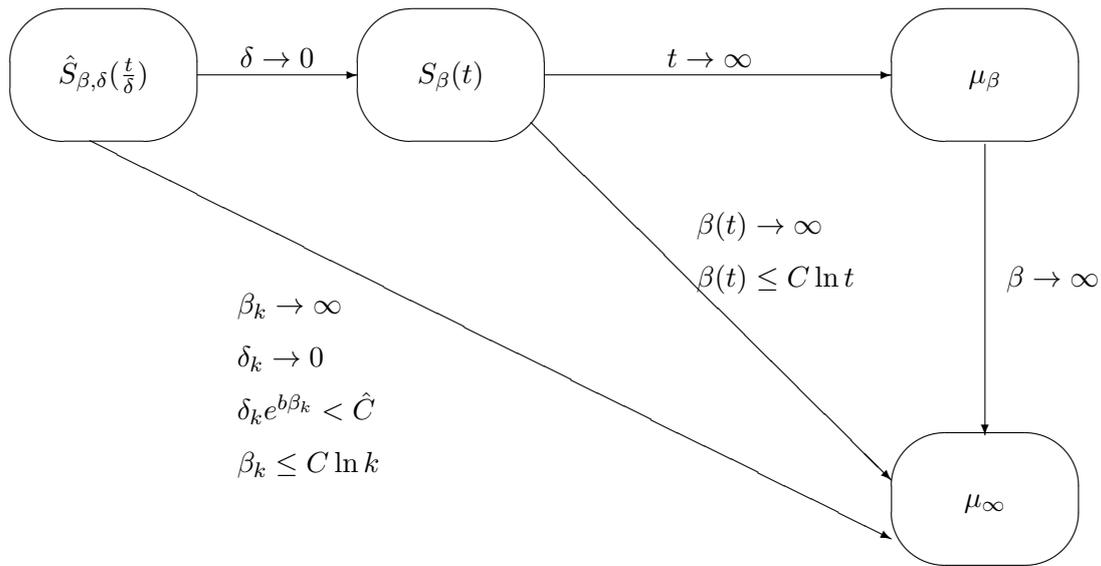
Здесь δ_k – шаг аппроксимации по времени, β_k – последовательность, задающая аннилинг, k – номер шага, на котором находится процесс, $\Delta y = \frac{1}{|V|}$ – объем, занимаемый одним пикселем изображения.

При этом все точки конфигурации γ могут погибнуть независимо друг от друга, и аналогично, в конфигурации независимо появляются новые точки. Таким образом, переходный оператор аппроксимирующей марковской цепи на шаге k имеет вид

$$(P_{\beta, \delta}^{(k)} f)(\gamma) = \sum_{\gamma_1 \subseteq \gamma} \prod_{x \in \gamma_1} \frac{1}{1 + \alpha_k(x, \gamma) \delta_k} \prod_{x \in \gamma \setminus \gamma_1} \frac{\alpha_k(x, \gamma) \delta_k}{1 + \alpha_k(x, \gamma) \delta_k} \quad (6)$$

$$\times \Theta^{-1}(\gamma_1) \sum_{m=0}^{\infty} \int_{V(\gamma_1)} \frac{(z \delta_k)^m}{m!} f(\gamma_1 \cup y_1 \dots \cup y_m) dy_1 \dots dy_m.$$

На основании формул (4), (5), (6) для переходных вероятностей создана компьютерная модель динамики рождения-гибели, приводящая к конфигурации γ^* . Ниже представлена схема, иллюстрирующая вышеупомянутые сходимости и динамики:



3. ФУНКЦИЯ ЭНЕРГИИ

Функция энергии $H : \gamma \mapsto \mathbb{R}$ каждой конфигурации $\gamma \in \Gamma(V)$ ставит в соответствие некоторое вещественное число, называемое энергией. Как следует из постановки задачи, см. (2), выбор функции энергии играет ключевую роль при гиббсовском подходе. С помощью энергии конфигурации мы измеряем, насколько данная конфигурация “близка к искомой”. Чем больше энергия конфигурации, тем дальше она от оптимальной. Конфигурация, состоящая только из точек, являющихся центрами всех голов на изображении, и есть “самая лучшая”, или оптимальная, для нас конфигурация. Именно на такой конфигурации достигается глобальный минимум функции $H(\gamma)$. Поэтому ясно, что выбор функции энергии $H(\gamma)$ – это очень важный и ответственный шаг всей работы. Заложив неточную или неправильную информацию в функцию $H(\gamma)$, мы получим на выходе конфигурацию γ^* , которая не обладает требуемыми свойствами. По сути, выражение для $H(\gamma)$ есть перевод на язык математики тех условий, которые мы предъявляем к искомой конфигурации γ^* .

Напомним, что мы имеем дело с некоторым набором точек или пикселей, который мы называем конфигурацией. Любой пиксель может войти в конфигурацию. При этом точки конфигурации мы рассматриваем как центры окружностей фиксированного радиуса. Наша цель – получить конфигурацию, состоящую из центров дисков окружающих головы всех людей на изображении. Ниже мы описываем функцию энергии $H(\gamma)$, которая используется в программе и объясняем наш выбор.

Энергия конфигурации представляет собой сумму двух слагаемых

$$H(\gamma) = H_{data}(\gamma) + \lambda H_{repulsion}(\gamma). \tag{7}$$

Первая часть $H_{data}(\gamma)$ – это *энергия данных*. Эта энергия определяется отдельно для каждой точки x , т.е. $H_{data}(\gamma) = \sum_{x \in \gamma} H_{data}(\{x\})$, и зависит только от данных, т.е. от исходного цифрового изображения, вторая часть полной энергии конфигурации – это *энергия взаимодействия точек в конфигурации*, в нашем случае, *энергия отталкивания*, λ – весовой коэффициент.

Для того, чтобы некоторая точка являлась центром лица, определенная область вокруг неё (которую мы приближаем диском) должна быть похожа на лицо, а то, что располагается за границами этой области, должно отличаться от внутренности (см. рис. 1). Особенность нашей задачи – это небольшой размер объектов по сравнению с размерами лиц в обычной задаче распознавания, где можно учитывать характерные признаки лица. Поэтому в нашей постановке задачи лицо отличается от других объектов, присутствующих на изображении, только его формой, размером и цветом. Первое слагаемое в (7), энергия данных, выбирается именно таким образом, чтобы формализовать условие, что минимальную энергию будет иметь тот диск, который хорошо совмещается (налагается) с некоторым лицом на изображении. Другими словами, внутренние пиксели такого диска должны быть схожи по цвету с выделяемыми объектами (в нашем случае, лицами людей в толпе), и кроме того, диск должен иметь хорошую контрастность с внешним окружением. Для этого мы рассматриваем значения цвета пикселей внутренней и внешней области как выборки трехмерной случайной величины. Мы подсчитываем эмпирическое среднее и эмпирическую дисперсию для внутренней и внешней областей каждой точки (в качестве внешней области рассматривается кольцо, окружающее внутренний диск); определенным образом сравниваем их и присваиваем каждой точке значение, отражающее то, насколько она похожа на центр лица: чем меньше эта величина, тем больше сходство.

Теперь формализуем все вышесказанное. Прежде всего, энергия данных $H_{data}(\gamma)$ представляет собой сумму энергий данных по всем точкам x конфигурации γ :

$$H_{data}(\gamma) = \sum_{x \in \gamma} H_{data}(\{x\}).$$

Энергия данных отдельной точки также распадается на два слагаемых: $H_{data}(\{x\}) = H_1(\{x\}) + \alpha H_2(\{x\})$, где α – весовой коэффициент.

Для того, чтобы определить $H_1(\{x\})$, нам понадобится ввести несколько вспомогательных величин. Области

$$D_1(x) = \{t \in V : \|t - x\| \leq r\} \quad \text{и} \quad D_2(x) = \{t \in V : r < \|t - x\| \leq R\}$$

задают соответственно внутренность диска с центром в точке x радиуса r и внешность этого диска (кольцо с внешним радиусом R). Как уже упоминалось выше, все объекты предполагаются фиксированного размера (см. рис. 1). Здесь $\|\cdot\|$ – евклидова норма на решетке.



Рис. 1. Области D_1 и D_2 .

Введем вектора $\mu_1, \mu_2, \sigma_1 \in \mathbb{R}^3$, соответствующие эмпирическому среднему и дисперсии внутренней области:

$$\mu_1(x) = \frac{\sum_{y \in D_1(x)} \vec{\omega}_y}{|D_1(x)|}, \quad \mu_2(x) = \frac{\sum_{y \in D_2(x)} \vec{\omega}_y}{|D_2(x)|},$$

$$\sigma_1^{(j)}(x) = \frac{\sum_{y \in D_1(x)} (\omega_y^{(j)})^2}{|D_1(x)|} - (\mu_1^{(j)}(x))^2, \quad j = 1, 2, 3, \quad |D_i(x)| = \sum_{y \in D_i(x)} 1, \quad i = 1, 2.$$

Воспользуемся евклидовой метрикой и вычислим расстояние между векторами μ_1 и μ_2 .

$$B(\mu_1(x), \mu_2(x)) = \sum_{j=1}^3 k_j (\mu_1^{(j)}(x) - \mu_2^{(j)}(x))^2,$$

где k_1, k_2, k_3 – весовые коэффициенты, определяемые освещением, качеством снимка и цветовой схемой, о которой подробнее будет сказано в следующем параграфе. Первое слагаемое энергии данных выглядит следующим образом:

$$H_1(x) = \begin{cases} 1 - \frac{B(\mu_1(x), \mu_2(x))}{\Theta}, & B(\mu_1(x), \mu_2(x)) \leq \Theta, \\ e^{-\frac{B(\mu_1(x), \mu_2(x)) - \Theta}{3B(\mu_1(x), \mu_2(x))}} - 1, & B(\mu_1(x), \mu_2(x)) > \Theta, \end{cases}$$

где Θ – пороговое значение контрастности, то есть такое значение $B(x)$, которым обладают наименее контрастные точки, тем не менее, являющиеся центрами лиц. Это значение мы определяем вручную, выбирая такие точки.

Для того, чтобы определить $H_2(x)$, введем вектор $\mu_s = \mu_1(x_{sample})$, где x_{sample} – пиксель, выбранный вручную и соответствующий центру типичного для данного изображения лица, то есть $\mu_s^{(1)}$ – средний цвет типичного лица. Зададим $H_2(x)$ следующим образом:

$$H_2(x) = m_1 \left(\mu_1^{(1)}(x) - \mu_s^{(1)} \right)^2 + m_2 \left(\mu_1^{(2)}(x) - \mu_s^{(2)} \right)^2 + m_3 \left(\mu_1^{(3)}(x) - \mu_s^{(3)} \right)^2 + m_4 |\sigma_1(x)|^2.$$

Здесь множители m_1, m_2, m_3, m_4 – это также весовые коэффициенты, изменяя которые, можно влиять на качество выделения. Как перспективу дальнейшей деятельности можно рассмотреть разработку вспомогательного алгоритма, который решал бы задачу поиска оптимальных значений этих множителей.

Из приведенных выше формул можно заметить, что $H_1(x)$ является членом, отвечающим за контрастность расположенного в x диска. Чем больше разница между средним цветом внутренности этого диска и средним цветом окружения, тем меньше $H_1(x)$ и тем больше вероятность, что точка x является центром лица и останется в конфигурации. Второе слагаемое $H_2(x)$ отвечает за цветовое сходство диска с центром в точке x с лицом, учитывая также его однородность, т.к. само по себе сходство может оказаться ошибочным из-за усреднения.

Также необходимо отметить, что определение энергии данных через множества D_1 и D_2 делает невозможным расчет $H_{data}(x)$ для пикселей, которые находятся близко к границе области V , поскольку точки из множеств D_1 и D_2 будут выходить за пределы V . На данном этапе мы пренебрегаем этим эффектом и не выделяем точки, находящиеся на границе.

Далее, энергия отталкивания возникает вследствие парного взаимодействия точек конфигурации. Как уже было сказано, при выделении лиц в плотной толпе возможны наложения объектов друг на друга. Это означает, что точки из предельной конфигурации γ^* могут располагаться на расстоянии, меньшем чем $2r$, то есть так, что окружности центрами которых они являются будут пересекаться. Однако, полного совпадения и сильного пересечения быть не должно. Для того, чтобы учесть эти условия для каждой пары точек конфигурации, мы определяем энергию отталкивания следующим образом:

$$\forall \{x, y\} \in \gamma^2, \quad H_{repulsion}(x, y) = \max \left(0, 1 - \frac{\|y - x\|}{2r} \right), \quad (8)$$

где r – фиксированный ранее радиус дисков. Формула (8) означает, что для точек, находящихся на расстоянии, большем, чем $2r$, отталкивания нет, а для точек, находящихся на более близком расстоянии друг от друга, отталкивание увеличивается при уменьшении этого расстояния.

В заключение этого раздела подытожим все вышесказанное. Наша дискретная модель динамики рождения-гибели порождает последовательность конфигураций пошагово. На каждом шаге новая конфигурация генерируется случайным образом на основе определенного распределения вероятностей. Это распределение вероятности определяется функцией энергии $H(\gamma)$, которая, в свою очередь, зависит от данных ($H_{data}(\gamma)$) и от расположения точек в конфигурации на предыдущем шаге ($H_{repulsion}(\gamma)$).

4. АЛГОРИТМ

В этом разделе мы рассмотрим программную реализацию и обсудим некоторые возможные отступления от модели.

Для обработки цветных изображений была выбрана цветовая модель HSV, представляющая цвет каждого пикселя как трехмерный вектор $\vec{\omega} \in R^3$ с координатами: цветовой тон (Hue), насыщенность (Saturation) и яркость (Value): $\omega_i \in [0, 1]$, $i = 1, 2, 3$. Для нас удобно использовать такое представление цвета, так как лица на изображении могут быть по-разному освещены и иметь разные оттенки. С помощью весовых коэффициентов k_1, k_2, k_3 мы увеличиваем вклад первой компоненты «цветовой тон» относительно остальных, что позволяет нам лучше задавать сходство объектов по цвету: $k_1 = 1$, $k_2 = k_3 = 0.1$.

- **Предварительная обработка (фильтрация изображения):** для каждого пикселя x вычисляется величина $\|\vec{\omega}_x - \vec{\mu}_S(1)\|$, где $\vec{\mu}_S \in R^3$ – средний цвет типичного лица. Если эта разница превосходит установленный порог, пиксель перекрашивается в черный цвет. Пример изображения после предварительной обработки:



Рис. 2. Результат обработки.

Как видно из рисунка 2 после предварительной обработки задача несколько упростилась, тем не менее осталось большое количество фрагментов, схожих по цвету с лицом, но не

подходящих по размеру. Далее в качестве данных мы рассматриваем изображение, полученное в результате фильтрации. На последующих этапах точки, перекрашенные в черный цвет, не рассматриваются для включения в конфигурации.

- **Вычисление энергии данных:** для каждого пикселя x , не перекрашенного в черный цвет, вычисляется значение $H_{data}(x)$ по описанной выше схеме.

Далее мы рассмотрим два варианта дальнейших действий. Первый – это продолжить действовать в соответствии с описанной выше стохастической схемой. Второй вариант – детерминированный, основан только на значениях энергии данных. Заметим, что H_{data} построена таким образом, что “лучшие” (максимально похожие на центры голов) точки доставляют ей минимум. Таким образом, если мы возьмем точки, в которых функция H_{data} принимает минимальное или близкие к минимальному значения, можно предположить, что полученная конфигурация будет близкой к γ^* . Это упрощение существенно облегчает задачу и ускоряет процесс, поскольку, именно динамика рождения гибели является наиболее сложной частью в вычислительном плане.

- **Цикл, моделирующий динамику:** В программе для моделирования динамики рождения-гибели используется цикл while. В качестве критерия останова используется условие, что все новые объекты, появившиеся на шаге рождения, сразу погибают на шаге гибели, и, следовательно, конфигурация не меняется.

На первом шаге конфигурация пуста и для каждой точки разыгрывается случайная величина попадания в конфигурацию по формуле (5). На последующих шагах точки конфигурации с вероятностью (4) либо “гибнут” (то есть исключаются из конфигурации), либо “выживают”. В то же время, точки, не принадлежащие конфигурации либо добавляются в конфигурацию согласно (5), либо нет. Также, внутри цикла каждый раз подсчитывается относительная энергия отталкивания $H_{repulsion}$. Ее невозможно посчитать заранее как H_{data} , поскольку она зависит от тех точек, которые на данном шаге входят в конфигурацию.

- **Детерминированный подход:** Итак, как уже было сказано, мы можем вычислить энергию данных для всех точек. Далее определяем конфигурацию $\hat{\gamma}^*$:

$$\hat{\gamma}^* = \{x \mid H_{data}(x) < \min(H_{data}) + P\},$$

где P – отступ от минимального значения, который позволяет контролировать, насколько близкие к минимуму точки мы берем в конфигурацию. Таким образом, мы обходимся без динамики рождения-гибели и пересчета энергии отталкивания. Но возникает проблема: если сделать отступ P очень маленьким, большинство лиц окажутся невыделенными, кроме самых “удачных” в том смысле, что они обладают наименьшей энергией данных. Если же увеличить P , то каждому лицу будет соответствовать несколько точек из его центра. Этот эффект неизбежен, так как в условиях низкого разрешения отдельных лиц невозможно “настроить” H_{data} так, чтобы каждому лицу соответствовала точно одна точка в центре. В нашем первом варианте эта проблема решается за счет энергии отталкивания.

Так как здесь нет энергии отталкивания, нам нужно в каждой группе выделенных точек оставить одну центральную, при этом количество групп заранее неизвестно. Таким образом, возникает задача кластеризации с неизвестным количеством центроидов. Это одна из открытых задач анализа данных, так как для качественной кластеризации необходимо знать количество кластеров или иную информацию, помимо координат объектов. Тем не менее, существуют методы, не требующие никаких параметров. В данной работе мы воспользовались одним из них, методом “Mean Shift”, подробнее см. [17]. В результате из конфигурации $\hat{\gamma}^*$ получается “прореженная” конфигурация $\tilde{\gamma}^*$, состоящая из центров кластеров.

5. РЕЗУЛЬТАТЫ

В этом разделе представлены результаты работы стохастического алгоритма на нескольких изображениях. Они сравниваются с результатами, полученными с применением детерминированного подхода.

Так как исследуемый алгоритм является в некотором смысле классификатором, его характеризуют ошибки первого и второго рода, то есть пропуски лиц и ложные срабатывания. Здесь мы приведем несколько фрагментов двух фотографий большой плотной толпы на которых продемонстрированы результаты выделения полученные разными методами.



Рис. 3. Митинг на проспекте Сахарова, 2012.



Рис. 4. Фрагмент № 1 рисунка 3. Обработка MBAD.



Рис. 5. Фрагмент № 1 рисунка 3. Обработка Mean Shift.

Визуальный анализ результатов выделения на обработанных фрагментах показал, что ошибка первого рода алгоритма MBAD составляет 8.6%, ошибка второго рода: 15.2%. Для алгорит-



Рис. 6. Фрагмент № 2 рисунка 3. Обработка МВАД.



Рис. 7. Фрагмент № 2 рисунка 3. Обработка Mean Shift.



Рис. 8. Митинг на Болотной площади, 2012.



Рис. 9. Фрагмент № 1 рисунка 8. Обработка MBAD.



Рис. 10. Фрагмент № 1 рисунка 8. Обработка Mean Shift.



Рис. 11. Фрагмент № 2 рисунка 8. Обработка MBAD.



Рис. 12. Фрагмент № 2 рисунка 8. Обработка Mean Shift.

ма, основанного на кластеризации, значения ошибок первого и второго рода составили 8.7% и 17% соответственно.

Общее число людей на рисунке 3 по алгоритму с использованием кластеризации составило 2203 человека, алгоритм MBAD выделил 2121 лицо.

Исходя из имеющихся результатов, можно сделать два вывода. Во-первых, выделение в обоих случаях происходит и происходит успешно, во-вторых, результаты стохастического подхода и алгоритма, основанного на новом детекторе и кластеризации, различаются довольно мало. Это говорит о том, что на данном этапе исследований энергия данных влияет на результат выделения сильнее, чем энергия отталкивания. Кроме того, на тех данных, которыми мы располагаем, время работы MBAD значительно превышает время кластеризации. Это в первую очередь может означать, что компьютерная реализация стохастической динамики требует другой среды программирования. В работе использовался язык программирования Matlab, поскольку на нем удобны и легкодоступны средства работы с изображениями, но, как известно, на Matlab медленно реализованы циклы. Поскольку в программе, моделирующей динамику рождения-гибели, цикл является основным механизмом, в дальнейших исследованиях будет сделан выбор в пользу другой среды программирования.

6. ЗАКЛЮЧЕНИЕ

В данной работе мы предложили стохастический алгоритм для задачи выделения лиц, основанный на стохастической динамике рождения-гибели. Результаты, продемонстрированные на рисунках предыдущего раздела, позволяют сделать вывод что алгоритм, успешно выделяющий целый ряд объектов, начиная от крон деревьев и птиц и заканчивая дорогами и

нейронами головного мозга, можно также приспособить для выделения лиц в плотной толпе людей, в случае, когда эффектом перспективы можно пренебречь.

Алгоритм МВАД можно также использовать в случае, когда на изображении исследуемые объекты имеют различные размеры, например, когда замечен эффект перспективы. Для этого достаточно несколько изменить математическую модель: вместо точечного поля следует рассмотреть маркированное точечное поле. В этом случае конфигурационное пространство модели будет другим: вместо дисков фиксированного радиуса будет множество дисков с переменным радиусом, изменяющихся в некоторых пределах, или, в зависимости от задачи, множество эллипсов с переменными осями. При этом радиус диска будет играть роль марки.

Отметим, что предложенный алгоритм пока требует определенной ручной настройки. Этот факт делает невозможным запуск программы на какой-либо базе изображений. Мы планируем усовершенствовать алгоритм таким образом, чтобы настройка параметров под каждое изображение происходила также в автоматическом режиме. Этого можно достичь, уменьшив количество параметров за счет изменения некоторых механизмов нашей модели.

Вместе с первой проблемой возникает необходимость автоматической оценки результата. Единственный вариант, который на сегодняшний день представляется возможным – это ручная разметка всех изображений базы, на которой происходит проверка, то есть на каждом изображении нужно отметить все области соответствующие лицам. Поскольку нам потребуется база, каждое изображение которой содержит порядка тысячи объектов, эта задача представляется чрезвычайно трудоемкой и ее решение – не простой вопрос.

Помимо стохастического алгоритма мы также предложили альтернативный подход к задаче выделения множественных объектов, основанный на минимизации энергии данных и последующей кластеризации методом Mean-Shift. Этот подход дает результат в среднем того же качества, что и стохастический алгоритм, но значительно быстрее. Мы планируем в следующей работе провести более детальное сравнение этих подходов к решению задачи выделения и протестировать предложенные алгоритмы на большой базе изображений.

СПИСОК ЛИТЕРАТУРЫ

1. Green, P.J., Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination, *Biometrika*, 1995, vol. 82, no. 4, pp. 711–732.
2. Descombes, X., Minlos, R.A., and Zhizhina, E., Object Extraction Using Stochastic Birth-and-Death Dynamics in Continuum, *J. Math. Imaging and Vision*, 2009, vol. 33 (3), p. 347.
3. Descombes, X. and Zhizhina, E., The Gibbs Fields Approach and Related Dynamics in Image Processing, *Condensed Matter Physics*, 2008, vol. 11, no.2, pp. 1–20.
4. Benedek, C., Descombes, X., and Zerubia, J., Building Extraction and Change Detection in Multitemporal Remotely Sensed Images with Multiple Birth and Death Dynamics, *Workshop on Applications of Computer Vision – WACV*, 2009, pp. 1–6.
5. Kulikova, M., Jermyn, I., Descombes, X., and Zerubia, J., A Marked Point Process Model with Strong Prior Shape Information for the Extraction of Multiple, Arbitrarily-shaped Objects, *Signal-Image Technology and Internet-Based Systems*, 2009, pp. 180–186.
6. Cariou, P., Descombes, X., and Zhizhina, E., A Point Process for Fully Automatic Road Network Detection in Satellite and Aerial Images, *Electronic Journal of Information Processes*, 2010, 10(3), pp. 247–256.
7. Viola, P. and Jones, M., Robust Real-time Object Detection, *Int. J. Computer Vision*, 2001.
8. Irshad, A. and Matthew, N.D., Multiple Human Tracking in High-density Crowds, *J. Image and Vision Computing Archive*, 2012, vol. 30, Issue 12, pp. 966–977.

9. M. Rodriguez, M., Laptev, I., Sivic, J., and Audibert, J.-Y., Density-aware Person Detection and Tracking in Crowds, *Proc. IEEE International Conf. on Computer Vision*, 2011.
10. Ge, W. and Collins, R.T., Marked Point Processes for Crowd Counting, *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June, 2009, pp. 2913–2920.
11. Desai, C., Ramanan, D., and Fowlkes, C., Discriminative Models for Multi-Class Object Layout, in *ICCV*, 2009.
12. Torralba, A., Contextual Priming for Object Detection, *IJCV*, 2003, 53(2).
13. Descombes, X. and Zhizhina, E., Double Annealing Regimes in the Multiple Birth-and-Death Stochastic Algorithms, *Markov Processes and Related Fields*, 2012, vol. 18, no 3, pp. 441–456.
14. Geman, S. and Geman, D., Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images, *IEEE Trans. Pattern Analysis and Machine Intelligence*, 1984, vol. 6, no. 6, pp. 721–741.
15. Haario, H. and Saksman, E., Simulated Annealing Process in General State Space, *Adv. Appl. Probab.*, 1991, vol. 23(4), pp. 866–893.
16. Van Lieshout, M.N.M., Stochastic Annealing for Nearest-Neighbour Point Processes with Application to Object Recognition, *Advances in Applied Probability*, 1994, vol. 26, no. 2, pp. 281–300.
17. Cheng Yizong, Mean Shift, Mode Seeking, and Clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1995, vol. 17 (8), pp. 790–799.

Detection of Blurred Objects on Color Images Using Stochastic Algorithms

D. Sidorchuk, E. Zhizhina

Nowadays every digital camera or modern mobile phone can solve the problem of face detection under the conditions of small number of people and short distances. The current research is focussed on extreme cases. Detecting and counting overlapping objects with blurred boundaries in high density swarms are still open problems of image analysis. To solve this problem we use stochastic MBAD algorithm that has already shown good results in other fields of computer vision. The work is devoted to the explanation of the basic principles of the algorithm and describes its adaptation to the face detection problem. In addition to the stochastic algorithm, we introduce here a new face detector for blurred objects based on the same energy function that we have used in the stochastic approach.

KEYWORDS: face detection, birth and death dynamics in continuum, stochastic modeling, annealing.