Композиции признаков для видеотрекинга при помощи фильтра частиц*

Е. А. Нижибицкий
nizhibitsky@cs.msu.ru
Москва, Факультет ВМК МГУ имени М. В. Ломоносова

Рассмотрены модели правдоподобия, основанные на композиции мер сходства извлекаемых из изображений признаков, которые широко используются для задачи отслеживания объектов на видео при помощи фильтра частиц. Предложены новые способы оптимального многократного извлечения признаков из различных регионов одного и того же изображения. Оптимизация при этом выполняется за счет построения интегральных изображений, впервые примененных в компьютерном зрении для признаков Хаара в алгоритме Виолы–Джонса, для других исследуемых признаков. Экспериментально показана возможность эффективного использования композиций групп признаков при неэффективности использования каждой группы в отдельности. С помощью рассмотренных композиций достигнуто качество трекинга, сравнимое с более сложными по своей структуре методами, основанными на построении ансамблей с помощью бустинга, и превышающее результаты схожей работы с применением метода каскадов.

Ключевые слова: *трекинг; фильтр частиц; интегральное изображение; гистограмма направленных градиентов; признаки Хаара; локальные бинарные шаблоны; композиция признаков*

Feature composition in video tracking using particle filters*

E. A. Nizhibitsky
Faculty of Computational Mathematics and Cybernetics, M. V. Lomonosov Moscow State University

This work considers the likelihood models based on similarity measures extracted from image features which are widely used in the field of video tracking using particle filters. New computationally optimal methods for multiple feature extraction from several regions of the same image are proposed. The optimization is performed by using integral images, first prominently used in computer vision within Viola–Jones object detection framework for Haar rectangles and for other studied features. It is experimentally demonstrated that feature compositions can be used even in the tasks where each of them is useless by itself. The performance achieved using the proposed compositions is greater than one in the similar study and comparable to the performance of more complicated models based on ensemble boosting.

Keywords: *tracking; particle filter; integral image; histogram of oriented gradients; HOG; Haar features; local binary patterns; LBP; feature composition*

1 Введение

Задача трекинга объектов на видео является частью таких прикладных областей, как построение систем видеонаблюдения, отслеживания дорожного трафика (в частности, наблюдения за определенными транспортными средствами в потоке) [1], создание интерфейсов человек–компьютер [2], программ для передачи и сжатия видео [3] и др.

*Работа выполнена при финансовой поддержке РФФИ, проект № 14-07-00965.

За последние годы было предложено множество успешных подходов по решению данной задачи [4], но многие из них накладывают свои ограничения на обрабатываемые данные — например, статичный фон и фиксированный ракурс [5], знание о типе наблюдаемого объекта [1] или даже наличие множества камер [6]. Одни подходы уделяют мало внимания вычислительной сложности, другие, наоборот, учитывают строгие ограничения по ресурсам, примером чего являются приложения в робототехнике [7].

Многие из них (см. обзор в [4]) опираются на использование фильтра частиц для приближения вероятностного распределения на положения объекта на видео с помощью частиц, или сэмплов, которым отвечают регионы на видео и те или иные дополнительные характеристики [8]. Для каждой частицы при этом необходимо подсчитывать ее вес, пропорциональный схожести данного региона с регионом для отслеживаемого объекта; следовательно, необходимо уметь выделять признаки из них.

Каждая из таких работ при введении упомянутых мер сходства опирается на свой ограниченный набор признаков, тогда как другие признаки не рассматриваются или по причине самостоятельной неэффективности в условиях каких-то возникающих на исследуемых видео сложностей (например, изменяющееся освещение), или из-за вычислительной сложности многократного выделения признаков для каждой частицы. В разных работах при этом одновременно могут (не)использоваться одни и те же признаки при схожей аргументации (см., к примеру, цветовые признаки и LBP (local binary patterns) в [1, 8]).

Целью данной работы является исследование возможности эффективного использования композиций признаков даже там, где каждый из них может быть неэффективен сам по себе, а также получение способов оптимального многократного извлечения этих признаков из различных регионов одного кадра видеоряда. Это позволит в некоторых задачах задействовать простые композиции «слабых» признаков, не прибегая к вычислительно более затратным. В качестве базовой модели используется модель трекинга из [8] без бустинга, которая дополняется моделями правдоподобия на основе изучаемых признаков.

2 Постановка задачи

Рассматривается задача трекинга объекта на фрагменте видео, где в каждом кадре под положением объекта понимается прямоугольный регион, наилучшим образом его описывающий, — для каждого номера кадра t есть истинное значение $X_t = (x, y, w, h)$, при этом X_0 считается заданным, тем самым осуществляется сопровождение заданного своим начальным положением объекта. Целью отслеживания в данной постановке является поиск приближения $\hat{X}_1, \ldots, \hat{X}_T$ для истинных значений $X_1, \ldots, X_T$ ($\hat{X}_0 = X_0$).

2.1 Используемая мера качества

Для того чтобы определять, насколько выделяемый регион похож на реальный регион, соответствующий отслеживаемому объекту, нужно учитывать не только то, насколько близок центр рассчитанного выделения к реальному центру отслеживаемого объекта, но и то, насколько велика разница между реальными и вычисленными размерами объекта. Для экспериментов в данной работе использовалась мера схожести регионов, предложенная в работе [8], определяющая долю пересечения двух регионов в их объединении (рис. 1).

Аналогично упомянутой работе будем считать, что доле перекрытия выше 33% соответствует правильное определение положения объекта, а итоговое качество наблюдения будем считать как процент кадров, на которых это правильное определение произошло:

$$J(\{\hat{X}_t\}_{t=1}^T, \{X_t\}_{t=1}^T) = \frac{1}{T}\sum_{t=1}^{T} \mathbb{I}[\text{overlap}\left(\hat{X}_t, X_t\right) \geqslant 0{,}33] \,.$$

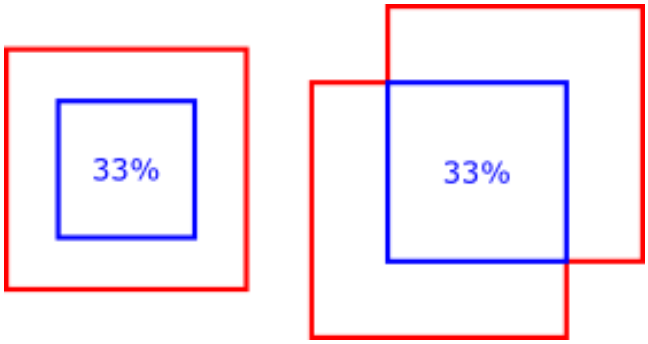


Рис. 1 Пересечение, соответствующее правильному определению положения объекта

3 Фильтр частиц для видеотрекинга

В этом разделе будет рассмотрено применение фильтра частиц для задачи видеотрекинга на основе модели, представленной в [8]. В последующих разделах будет подробнее освещено использование модели правдоподобия алгоритма, которая основывается на комбинировании методов оценивания схожести отдельных регионов изображения с отслеживаемым объектом.

Распределение $p(x_t)$ возможных положений отслеживаемого объекта на i-м кадре видео приближается набором $S_t = \{s_t^i\}_{i=1}^N$ (далее N=1000) взвешенных частиц $s_t^i = (x_t^i, \pi_t^i)$:

$$\hat{p}(x_t) = \sum_{i=1}^{N} \pi_t^i \delta_{x_t^i}(x_t).$$

В работе в качестве состояния частицы рассматривается вектор $(x, y, v_x, v_y, s)^\top$, учитывающий положение и скорость перемещения рамки, содержащей отслеживаемый объект, а также масштаб относительно ее первоначальных размеров при инициализации (считаем, что пропорции объекта не изменяются). При инициализации начальное положение вместе с рамкой (x, y, w, h) передается алгоритму, а веса частиц принимаются равными между собой ($\pi_0^j = 1/N$), начальные скорости (v_x, v_y) получаются из нормального распределения.

Для каждого нового кадра получается новый набор на основе алгоритма фильтра частиц с промежуточным ресэмплингом с учетом важности (веса) каждой частицы (Sampling Importance Resampling Particle Filter):

Алгоритм 1 Фильтр частиц для видеотрекинга (SIR PF)

Вход: S_{t-1}
Выход: S_t
  для $i = 1$ **to** N
    получить $x_t^i \sim p(x_k \,|\, x_{t-1}^i)$ *// модель движения*
    вычислить $\pi_t^i = p(Z_t \,|\, X_t = x_t^i, Z_0, Z_1, \ldots, Z_{t-1})$ *// модель правдоподобия*
  вычислить $w = \Sigma_{i=1}^N \pi_t^i$
  для $i = 1$ **to** N
    вычислить $\pi_t^i = w^{-1}\pi_t^i$ *// нормализация весов*
  для $i = 1$ **to** N
    получить $\hat{x}_t^i \sim \{p(x_k = x_t^j) = \pi_t^j, j = 1, \ldots, N\}$ *// ресэмплинг*
  return $\{(\hat{x}_t^i, N^{-1})\}_{i=1}^N$

Таким образом на каждом шаге на основе имеющихся частиц получаются новые с помощью моделирования их перемещения, затем оценивается правдоподобие нового состояния каждой частицы. В качестве промежуточного этапа перед новым шагом вместо частиц с различными весами получаются частицы с одинаковыми весами, где их состояния будут выборкой нужного размера из дискретного распределения на предыдущих состояниях с вероятностями, пропорциональными их весам.

Модель движения в рассматриваемых экспериментах учитывает имеющиеся скорости частиц для определения новых координат. Сами же скорости вместе с масштабом рамки для каждой частицы получаются на основе зашумления предыдущих значений нормальным распределением:

$$\begin{aligned} v_{x,t} &= v_{x,t-1} + N(0, \sigma_x^2); \\ v_{y,t} &= v_{y,t-1} + N(0, \sigma_y^2); \\ x_t &= x_{t-1} + v_{x,t}; \\ y_t &= y_{t-1} + v_{y,t}; \\ s_t &= s_{t-1} + N(0, \sigma_s^2). \end{aligned}$$

3.1 Модель правдоподобия

Чтобы вычислить вес каждой частицы, нужно оценить правдоподобие наблюдения, отвечающего ей, т. е. оценить схожесть региона, отвечающего частице, с шаблоном — таким же регионом для отслеживаемого объекта. Для этого достаточно извлекать признаки из регионов изображения и сравнивать их между собой. Далее под шаблоном также будут пониматься значения признаков для целевого объекта.

В данной работе рассматривались признаки, зарекомендовавшие себя в задаче отслеживания объектов, моделирующие представления объекта с разной стороны, а значит, подходящие для различных сложностей в исследуемых видео — одни модели используют цветовые признаки, другие же моделируют текстуру, контур или иные характеристики объекта. Для набора рассматриваемых признаков $\{f\}$ определялись меры схожести $\{\rho_f\}$ с шаблонами для реального объекта.

Распределение правдоподобия на основе одной метрики $\rho(\cdot, \cdot)$ можно получить по формуле:

$$p(Z_t|x_t) \propto \exp\left\{-\frac{\rho^2(\hat{h}_f, h_f(x_t))}{\lambda}\right\},$$

где $\hat{h}_f$ — шаблон; $h_f(x_t)$ — признаки для региона изображения, соответствующего состоянию x_t частицы s_t; λ — параметр, подбираемый отдельно для каждой пары признака и соответствующей ему метрики.

Общее правдоподобие наблюдения можно посчитать как произведение правдоподобий по каждому признаку:

$$p(Z_t|x_t) \propto \Pi_f \exp\left\{-\frac{\rho_f^2(\hat{h_f}, h_f(x_t))}{\lambda_f}\right\}.$$

В следующих разделах перейдем к описанию предлагаемых к использованию признаков вместе с оптимизациями для их многократного выделения.

4 Используемые признаки

4.1 Цветовые гистограммы

В качестве первой группы признаков рассматривались простые поканальные гистограммы для каждого из трех каналов RGB-изображения. Для оптимизации в дальнейшем эти значения в каждом канале объединялись в 8 *корзин* из 32 значений интенсивности. Три группы корзин, объединенные между собой и затем нормализованные, образуют вектор-признак из 24 значений. Метрика сходства двух гистограмм при этом определялась на основе Евклидовой метрики:

$$\rho_{\text{hist}}(\hat{h}, h(\mathbf{x_t})) = \sqrt{\sum_{i=1}^{24}(\hat{h}_i - h_i)^2}\,,$$

где $\hat{h}$ — шаблон; $h(x_t)$ — гистограмма для региона x_t, отвечающего частице s_t.

4.2 Признаки Хаара

Признаки Хаара впервые были описаны в [9], где они использовались в алгоритме для распознавания лиц, и на сегодняшний день применяются во многих алгоритмах классификации, так как обладают большей дискриминативной способностью, чем значения пикселей сами по себе. Для получения признаков каждый подрегион разбивается дополнительно на условно светлые и темные области, состоящие из одного или нескольких прямоугольников (рис. 2), для каждой из которых затем вычисляется среднее значение по каналам.

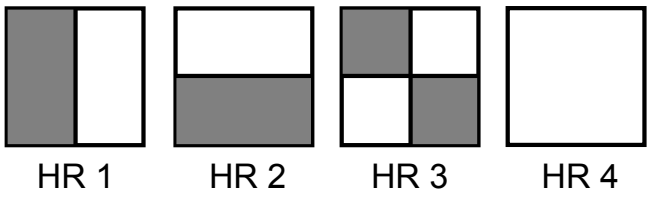


Рис. 2 Четыре признака Хаара для извлечения средних значений по каналам

Чтобы вычислить разницу между признаками для региона, отвечающего частице, и шаблоном, можно также воспользоваться Евклидовой метрикой для векторов из разностей значений в светлых и темных областях:

$$\rho_{\text{hr}}(\hat{c}, c(x_t)) = \sqrt{\sum_{v=1}^{4}((\hat{\text{red}}_v - \text{red}_{(x_t,v)})^2 + (\hat{\text{green}}_v - \text{green}_{(x_t,v)})^2 + (\hat{\text{blue}}_v - \text{blue}_{(x_t,v)})^2)}\,,$$

где $\hat{c}$ — цветовая информация из шаблона; $c(x_t)$ — цветовая информация для частицы с состоянием x_t; v — один из типов признаков, изображенных на рис. 2.

4.3 Гистограммы направленных градиентов

Признаки на основе цветов подходят для многих задач трекинга, даже когда происходят частичные наложения. Тем не менее, они плохо себя показывают в ситуации, когда на фоне присутствуют похожие цвета. Было предложено множество других типов признаков для использования вместе с цветовыми. В [10] показали, что комбинация цветовой модели

вместе с моделью контуров позволяет получить более быстрое и стабильное отслеживание объекта.

Для получения информации о контурах предлагается использовать гистограммы направленных градиентов (Histogram of Oriented Gradients, HOG). По своей природе они устойчиво себя ведут в условиях изменения освещенности и в случае схожести фона и объекта по цветовой модели. Для нахождения границ необходимо перевести RGB-изображение в градации серого, а затем вычислить операторы Собеля K_x и K_y [11]:

$$G_x(x,y) = K_x * I(x,y), G_x(x,y) = K_y * I(x,y) \text{ (под } * \text{ понимается свертка).}$$

Тогда сила (резкость перехода) и ориентация границы вычисляются по формулам:

$$\begin{aligned} S(x,y) &= \sqrt{G_x^2(x,y) + G_y^2(x,y)}; \\ \theta &= \arctan\left(\frac{G_y(x,y)}{G_x(x,y)}\right). \end{aligned}$$

Чтобы избавиться от шума, можно применить порог T к значению $S(x,y)$ (используем $T = 100$). Затем границы распределяются по K корзинам в соответствии с их направлениями, после чего значения их сил суммируются, и получается нужная нам гистограмма. В оригинальной работе [12] лучше всего показало себя $K = 9$. Схожесть между шаблоном и гистограммой для каждой частицы вычисляется аналогично предыдущим признакам с помощью Евклидовой метрики для векторов из корзин.

4.4 Локальные бинарные шаблоны

Другим примером получения информации о структуре региона изображения является извлечение локальных бинарных шаблонов, которые в некотором смысле характеризуют текстуру изображения в каждой конкретной точке, для чего и были описаны впервые в [13].

Главная идея данного метода состоит в извлечении локальной структуры путем сравнения интенсивности каждого пикселя с его соседями — для каждого соседа получается число, которое будет равно 1, если интенсивность соседа больше рассматриваемого центрального пикселя, и 0 — в противном случае. Если объединить полученные значения по часовой стрелке, то текстуру в окрестности каждого пикселя будет описывать вектор из 0 и 1 вроде 00010011, который и называется локальным бинарным шаблоном. Полученные вектора можно просуммировать по всем пикселям региона и, пронормировав, рассматривать их как гистограмму, определяющую текстурную характеристику всего региона. Схожесть между шаблоном и гистограммой для каждой частицы вычисляется аналогично предыдущим признакам.

5 Оптимизация многократного выделения признаков

Для начала стоит рассмотреть **интегральные изображения**, которые лежат в основе одних из используемых признаков (Хаара), а также будут играть значительную роль в оптимизации подсчета других. Затем будет показано, как можно оптимизировать многократный подсчет всех рассмотренных ранее признаков для различных регионов одного изображения.

5.1 Интегральные изображения

Чтобы получить интегральное изображение I на основе исходного F, для каждого пикселя необходимо вычислить значение по формуле:

$$I(x,y) = F(x,y) + I(x-1,y) + I(x,y-1) - I(x-1,y-1),$$

где $I(x,-1) = I(-1,y) = 0$. А это, очевидно, можно сделать за один проход по результирующему изображению с помощью динамического программирования.

После того как было получено интегральное изображение, для подсчета суммы интенсивностей для прямоугольника с верхним левым углом (x_1, y_1) и нижним правым углом (x_2, y_2) нужно воспользоваться формулой:

$$\sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} F(x,y) = I(x_2,y_2) - I(x_2,y_1-1) - I(x_1-1,y_2) + I(x_1-1,y_1-1)\,.$$

Данное выражение эквивалентно подсчету суммы для региона D изображения F с помощью вычисления $(A+B+C+D)-(A+B)-(A+C)+A$ для изображения I.

5.2 Цветовые гистограммы

Для оптимизации подсчета цветовых гистограмм для регионов на основе значения каждого канала всего текущего кадра видео создается 8 бинарных изображений, в каждом из которых значения будут характеризовать попадание интенсивности в нужную корзину гистограммы. Таким образом, каждое значение интенсивности цветового канала «оставит след» ровно на одном из 8 изображений, которые и будут исходными для получения интегральных изображений. Затем уже на основе этих интегральных изображений можно производить оптимальный расчет гистограмм для любых регионов в кадре — каждый элемент вектора гистограммы получается на основе соответствующих интегральных изображений за несколько простейших операций.

5.3 Признаки Хаара

Для признаков Хаара все гораздо проще — нам требуется создать только три интегральных изображения на основе цветовых каналов изображения из текущего кадра видео, с помощью которых для каждого прямоугольника Хаара каждого региона, отвечающего частице, подсчет нужных характеристик будет производиться за несколько простейших операций. Именно для этого вида признаков интегральные изображения впервые были использованы для оптимизации в [9].

5.4 Гистограммы направленных градиентов

Для эффективного вычисления гистограммы направленных градиентов подобно случаю «обычных» гистограмм для всего изображения можно построить K интегральных изображений на основе K исходных, каждое из которых вместо значений интенсивностей будет содержать либо значение силы границы, если направление совпадает с одним из девяти имеющихся, либо 0 в противном случае — т. о., все значения мощностей границ разойдутся по 9-ти исходным изображениям. Элементы вектора гистограммы региона затем получаются на основе сумм значений интенсивности в регионе для каждого из полученных исходных изображений, которые вычисляются за константное время на основе интегральных изображений.

5.5 Локальные бинарные шаблоны

Так же как и в случае с гистограммами направленных градиентов, значения для локальных бинарных шаблонов можно разделить на 8 отдельных корзин, для каждой из которых построить сначала исходное изображение, а затем интегральное, и с помощью рассмотренных выше приемов для каждого региона получать нужные гистограммы за константное время.

5.6 Теоретические результаты по ускорению

С помощью описанных выше оптимизаций удается избежать затратного многоразового извлечения признаков из регионов для каждой частицы. В табл. 1 приведены оценки вычислительных затрат на выделение N указанных признаков из региона $h \times w$ изображения размера $H \times W$.

Так, к примеру, для регионов, занимающих 10% площади кадра, при оптимизированном подсчете признаков Хаара получается 100-кратная экономия по вычислительным ресурсам при использовании 1000 частиц.

Таблица 1 Теоретические оценки сложности

Группа признаков	Без оптимизации	С оптимизацией
Интегральное изображение	O(Nwh)	O(WH)
Цветовые гистограммы	O(Nwh)	O(WH)
Гистограммы направленных градиентов	O(Nwh)	O(WH)
Локальные бинарные шаблоны	O(Nwh)	O(WH)

6 Вычислительные эксперименты

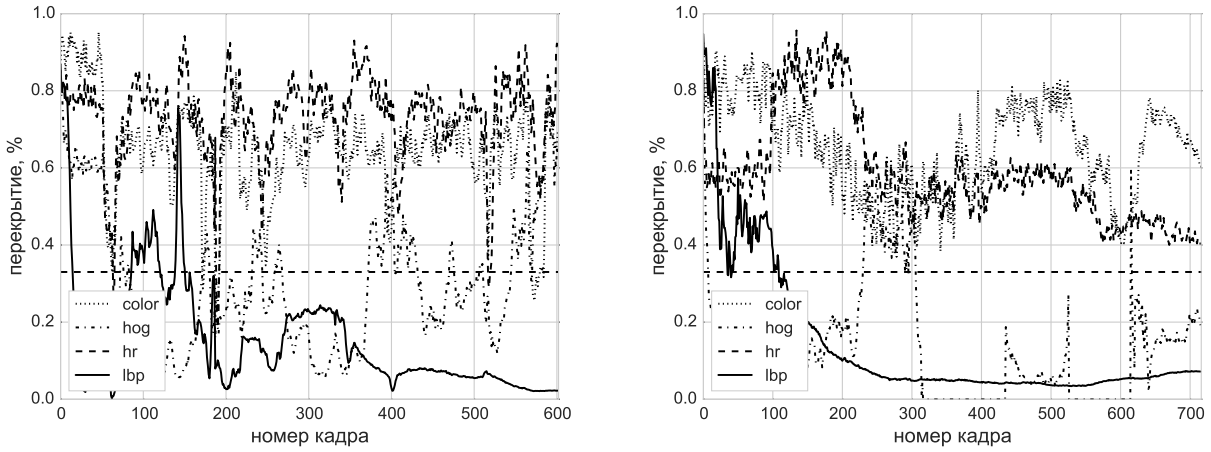
Все рассмотренные алгоритмы с учетом различных признаков были реализованы на языке `C++`. Для высокоуровневой работы с изображениями и видео использовалась библиотека `OpenCV` — в частности, с помощью нее происходила загрузка видео и разбор по кадрам на отдельные изображения, для которых в свою очередь применялись встроенные функции для операторов Собеля (`cv::Sobel`), подсчета интегральных изображений (`cv::integral`) и проверки на попадание в нужный интервал (`cv::inRange`).

6.1 Данные для экспериментов

Для вычислительных экспериментов использовался набор данных BoBoT, содержащий около десятка видео размера 320×240, каждое из которых отвечает тем или иным сложностям, возникающим при трекинге объектов, как-то: сложный неоднородный фон, изменяющееся освещение, перекрытие объектов, сильные перемещения объекта и/или камеры. Он доступен для скачивания на сайте одного из авторов работы [8] (`http://www.iai.uni-bonn.de/~kleind/tracking/`), где также предоставлен исходный код программы на `Java` для оценки качества на основе файлов истинной разметки и разметки, получаемой алгоритмами.

6.2 Результаты экспериментов

В первой части экспериментов были рассмотрены алгоритмы, использующие только одну характерную группу признаков. На рис. 3 изображены примеры графиков с распределением качества по кадрам для алгоритмов, которые для подсчета правдоподобия используют гистограммы цветов (`color`), гистограммы направленных градиентов (`hog`), прямоугольники Хаара (`hr`) или локальные бинарные шаблоны (`lbp`). Горизонтальными линиями отображен порог в 33%, по которому определяется успешность отслеживания объекта в данный момент времени. Из приведенных графиков можно сделать выводы, что самодостаточными для трекинга признаками являются цветовые гистограммы и прямоугольники Хаара. Лишь в некоторых видео значимый результат также показывали гистограммы градиентов. Использование локальных бинарных шаблонов видится осмысленным только в композиции с другими признаками.



(а) Видео A (ball) (б) Видео G (rubick's cube)

Рис. 3 Графики качества для алгоритмов на основе одной группы признаков

При использовании двух групп признаков `color` и `hr` значимые отличия в результатах проявились в анализе двух видео. На видео B с кружкой на очень пестром и разнообразном по текстуре фоне оба алгоритма на основе этих признаков теряют на некоторое время цель, но в разное время — один алгоритм находит похожую на кружку синюю сплошную часть доски, другой видит похожие с шаблонными перепады в цветах на графиках и фотографиях на доске. На видео H признаки Хаара быстро находят схожую по перепадам область и там же и остаются, тогда как цветовые гистограммы находят объект только при фиксированном типе освещения.

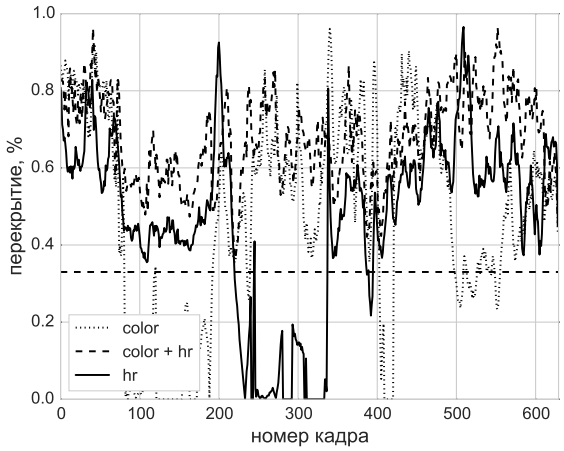
Так как оба типа признаков имеют схожую вычислительную сложность, было решено сравнить их качество по отдельности с композицией, а затем исследовать, как улучшает качество композиции добавление двух оставшихся «несамодостаточных» признаков.

Использование композиции позволяет нам брать лучшее от обеих групп признаков, что подтверждается экспериментами — на видео B при использовании композиции предсказание никогда не уходит от отслеживаемой кружки к схожим частям фона, что позволяет достичь 100%-ного качества. На видео E композиция лучше справляется с перекрытием объекта и не захватывает перекрывающий объект в качестве предполагаемого (рис. 4). Самый сильный же эффект от использования композиции достигается на видео H — наблюдается 100%-ное качество трекинга с постоянным пересечением с реальной областью на уровне 80%, в то время как каждый признак сам по себе совсем не справлялся с трекингом на данном видео (рис. 5).

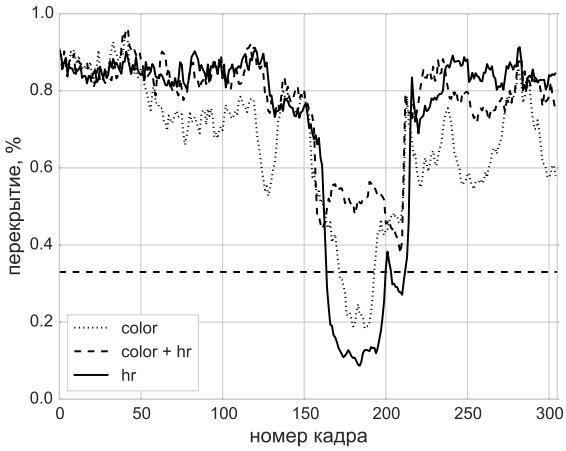
После добавления к изученной композиции «структурных» признаков, учитывающих структуру и текстуру объекта, на видео C стабильно лучше себя показали алгоритмы с добавлением LBP-признаков, что привело к очень высокому качеству трекинга для такой сложной задачи со значительным движением камеры и изменением масштаба объекта. На видео G добавление HOG уменьшило пересечения в конце где-то на четверть, что привело к вылету из минимальной зоны 33%-ного качества (рис. 6).

7 Заключение

В работе изучены способы выделения признаков из изображений для задачи трекинга объектов на видео, а также приведены способы их оптимального многоразового подсчета

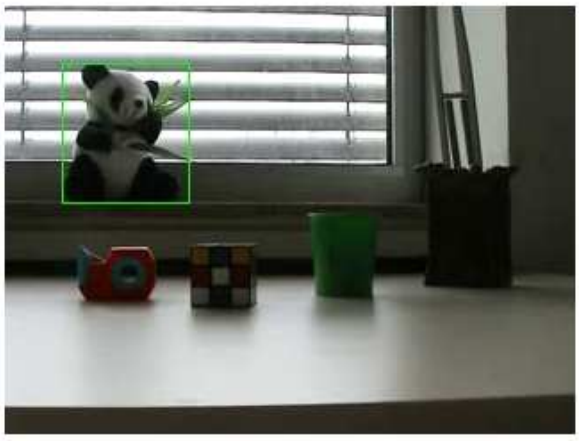


(а) График качества для видео B

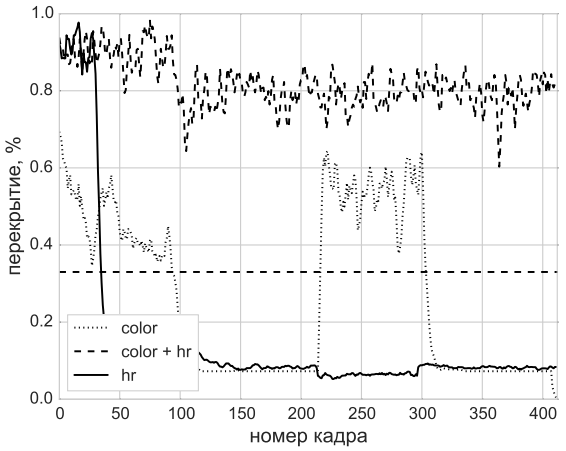


(б) График качества для видео E

Рис. 4 Пример объединения двух признаков в композицию.



(а) Кадр из видео H



(б) График качества для видео H

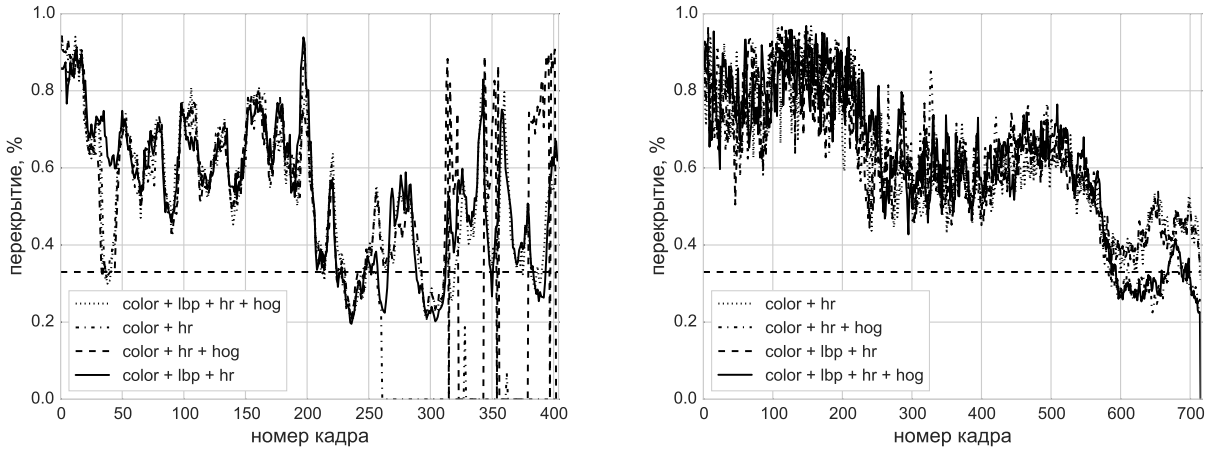
Рис. 5 Наиболее выраженный эффект от объединения двух признаков в композицию.

на основе обобщенного применения интегральных изображений, что позволило построить более богатые алгоритмы с использованием композиций рассмотренных признаков.

Экспериментально показано, что использование композиций признаков позволяет получить идеальные результаты при отслеживании положения сопровождаемого объекта на видео даже там, где алгоритмы на основе каждого из признаков в отдельности с этой задачей справиться не могли (рис. 5).

Из проведенных экспериментов можно также заключить, что наиболее универсальным в рамках рассматриваемых видео оказался композиционный алгоритм, основанный на использовании цветовых интегральных признаков, цветовых гистограмм и локальных бинарных шаблонов. Тем не менее простая комбинация из двух цветовых групп признаков также дает высокие результаты, компенсируя недостатки каждой группы в отдельности.

В табл. 2 приведено сравнение результатов экспериментов для всех рассмотренных алгоритмов на вышеупомянутом наборе данных — полужирным выделены наилучшие



(а) Видео C (juice) (б) Видео G (rubick's cube)

Рис. 6 Графики качества после добавления «структурных» признаков

Таблица 2 Сравнительная таблица качества всех рассмотренных алгоритмов

Группа признаков	A	B	C	D	E	F	G	H	I
`color`	0,966	0,732	0,252	0,947	0,931	0,556	**1,000**	0,441	0,719
`hr`	0,986	0,804	0,475	0,977	0,855	0,439	**1,000**	0,084	0,649
`hog`	0,367	0,184	0,178	0,714	0,200	0,569	0,117	0,415	0,074
`lbp`	0,102	0,081	0,091	0,114	0,347	0,256	0,146	0,038	0,066
`hr + color`	0,993	**1,000**	0,564	**0,991**	**1,000**	0,602	0,997	**1,000**	0,779
`hr + color + lbp`	**0,998**	**1,000**	0,836	**0,991**	**1,000**	**0,613**	**1,000**	**1,000**	**0,782**
`hr + color + hog`	0,996	0,992	0,764	**0,991**	**1,000**	**0,613**	0,853	**1,000**	0,770
`hr + color + lbp + hog`	0,996	0,992	**0,863**	0,990	0,996	**0,613**	0,863	**1,000**	0,772

результаты для каждого видеоряда. Напомним, что под качеством трекинга видео понимается доля кадров с достаточно сильным перекрытием между истинной и предсказанной рамкой, содержащей отслеживаемый объект (качеству 1,0 соответствует исход, при котором объект всегда находится в «поле зрения» модели).

С помощью рассмотренных композиций признаков достигнуто качество трекинга, сравнимое с более продвинутыми методами, основанными на построении сложных ансамблей с помощью бустинга [8], и превышающее результаты схожей работы [1] с использованием метода каскадов.

Литература

[1] *Samuelsson O.* Video tracking algorithm for unmanned aerial vehicle surveillance. Master's Degree Project at KTH Electrical Engineering. Stockholm, 2012.

[2] *Bradski G. R.* Computer vision face tracking for use in a perceptual user interface // Intel Technology J., 1998.

[3] *Vieux W. E., Schwerdt K., Crowley J. L.* Face-tracking and coding for video compression. Lecture notes in computer science ser., 1999.

[4] *Yilmaz A., Javed O., Shah M.* Object tracking: A survey // ACM Comput. Surveys, 2006. Vol. 38. No. 4.

[5] *Li H., Xiong S., Duan P., Kong X.* Multitarget tracking of pedestrians in video sequences based on particle filters // Advanced MultiMedia, 2012. Vol. 2012.

[6] *Xu M., Orwell J., Jones G.* Tracking football players with multiple cameras // IEEE Conference (International) on Image Processing Proceedings. — Los Alamitos, CA, USA: IEEE Computer Society Press, 2004. P. 2909–2912.

[7] *Fox D., Thrun S., Dellaert F., Burgard W.* Particle filters for mobile robot localization // Sequential Monte Carlo methods in practice. — New York, NY, USA: Springer Verlag, 2000.

[8] *Klein D A., Schulz D., Frintrop S., Cremers A. B.* Adaptive real-time video-tracking for arbitrary objects // IEEE Conference (International) on Intelligent Robots and Systems (IROS) Proceedings, 2010. P. 772–777.

[9] *Viola P., Jones M.* Rapid object detection using a boosted cascade of simple features // Conference on Computer Vision and Pattern Recognition Proceedings, 2001. P. 511–518.

[10] *Isard M., Blake A.* Icondensation: Unifying low-level and high-level tracking in a stochastic framework // 5th European Conference on Computer Vision Proceedings, 1998. Vol. I. P. 893–908.

[11] *Sobel I., Feldman G.* A 3×3 isotropic gradient operator for image processing. A talk at the Stanford Artificial Project. — Stanford, CA, USA, 1968.

[12] *Dalal N., Triggs B.* Histograms of oriented gradients for human detection // Conference on Computer Vision and Pattern Recognition Proceedings, 2005. P. 886–893.

[13] *Pietikäinen M., Ojala T., Xu Z.* Performance evaluation of texture measures with classification based on kullback discrimination of distributions // 12th IAPR Conference (International) on Pattern Recognition, 1994. P. 582–585.

References

[1] Samuelsson, O. 2012. Video tracking algorithm for unmanned aerial vehicle surveillance. Master's Degree Project at KTH Electrical Engineering. Stockholm.

[2] Bradski, G. R. 1998. Computer vision face tracking for use in a perceptual user interface. *Intel Technology J.*

[3] Vieux, W. E., K. Schwerdt, and J. L. Crowley. 1999. *Face-tracking and coding for video compression.* Lecture notes in computer science ser.

[4] Yilmaz, A., O. Javed, and M. Shah. 2006. Object tracking: A survey. *ACM Comput. Surveys* 38(4).

[5] Li, H., S. Xiong, P. Duan, and X. Kong. 2012. Multitarget tracking of pedestrians in video sequences based on particle filters. *Advanced MultiMedia* 2012.

[6] Xu, M., J. Orwell, and G. Jones. 2004. Tracking football players with multiple cameras. *IEEE Conference (International) on Image Processing Proceedings.* Los Alamitos, CA: IEEE Computer Society Press. 2909–2912.

[7] Fox, D., S. Thrun, F. Dellaert, and W. Burgard. 2000. Particle filters for mobile robot localization. *Sequential Monte Carlo methods in practice.* New York, NY: Springer Verlag.

[8] Klein, D A., D. Schulz, S. Frintrop, and A. B. Cremers. 2010. Adaptive real-time video-tracking for arbitrary objects. *IEEE Conference (International) on Intelligent Robots and Systems (IROS) Proceedings.* 772–777.

[9] Viola, P., and M. Jones. 2001. Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition Conference Proceedings.* 511–518.

[10] Isard, M., and A. Blake. 1998. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. *5th European Conference on Computer Vision Proceedings* I:893–908.

[11] Sobel, I., and G. Feldman. 1968. A 3x3 isotropic gradient operator for image processing. A talk at the Stanford Artificial Project.

[12] Dalal, N., and B. Triggs. 2005. Histograms of oriented gradients for human detection. *Conference on Computer Vision and Pattern Recognition Proceedings.* 886–893.

[13] Pietikäinen, M., T. Ojala, and Z. Xu. 1994. Performance evaluation of texture measures with classification based on kullback discrimination of distributions. *12th IAPR Conference (International) on Pattern Recognition Proceedings.* 582–585.