

Использование Радон и Фурье преобразований растровых изображений для описания и отслеживания заданных объектов

Е. А. Новиков, М. А. Падалко
eugen@novikov.de, padalkom@gmail.com

Институт прикладной математики и информационных технологий БФУ им. И. Канта,
Калининград, Россия

Как правило, существующие на сегодняшний день алгоритмы описания и идентификации объектов нацелены на решение задач распознавания определенного типа объектов в заданных условиях. Однако поиск универсального или более обобщенного подхода к решению данной задачи остается интересной проблемой с точки зрения академических исследований и перспективным с точки зрения практической реализации. Предлагаемый подход позволяет производить идентификацию изображений объектов по широкому спектру признаков. Метод представлен в виде общего описания алгоритма и результатов экспериментальной проверки его эффективности. Основная задача разработки метода — быстрая и качественная обработка графических данных в виде динамических изображений или видеопотоков. Доступные для сравнения методы используются преимущественно для поиска объектов в статических изображениях, в то время как авторский метод в первую очередь нацелен на работу с видеопотоками. Общедоступных видеоматериалов и данных по их обработке аналогичными методами для сравнительного анализа на момент написания статьи не найдено. Рассматриваемый метод предлагает новый способ получения набора ключевых признаков образа и функцию для их сравнения. Он основывается на применении комбинации классических методов прямого преобразования Радона к матрице изображения, одномерного преобразования Фурье к полученным интегральным проекциям и статистического анализа интегральных коэффициентов Фурье, рассматриваемых в качестве основных дескрипторов объектов изображения.

Ключевые слова: *компьютерное зрение; машинное зрение; анализ изображения; поиск заданного объекта; мониторинг видео*

DOI: 10.21469/22233792.1.13.05

The use of Radon and Fourier transformations of raster images for description and tracking of predefined objects

E. A. Novikov and M. A. Padalko

Immanuel Kant Baltic Federal University, Institute of Applied Mathematics
and Information Technologies, 14 A. Nevskogo st., Kaliningrad, Russia

As a rule, currently existing algorithms for describing and objects identification are aimed at solving the problem of definite types of objects in desired condition. However, search for all-in-one or more general approach to it is still quite interesting in the context of academic researches and from the perspective of implementation. Proposed approach allows the object identification on a wide range of characteristics. The method is presented in the form of a general algorithm description and results of the experimental test of its efficiency. The main task of method development is fast and quality processing of graphical data as dynamic images or video streams. Available for comparison methods are mainly used for object search in freeze-frame images while the presented method is primarily aimed at working with video stream. There are no generally available videos or data on their processing with similar methods

for comparative study as of this paper. Considered method allows one a new way of getting a key features set of the images and a function for their comparison. It is based on a combination of classical methods of direct Radon conversion of image matrix, one-dimensional Fourier conversion to the corresponding projections, and statistical analysis of integral Fourier coefficients, considered as objects features of images.

Keywords: *computer vision; machine vision; image analysis; object detection; video monitoring*

DOI: 10.21469/22233792.1.13.05

1 Введение

На сегодняшний день абсолютное большинство методов и технических средств обработки изображений предназначены для решения узкого класса прикладных задач и относительно надежно справляются с ними в рамках своего класса, однако в условиях среды с организованными помехами надежность существующих методов резко снижается. При этом временные затраты на обработку и принятие решения практически всегда значительно превышают требуемые в рамках поставленной задачи.

Стандартная схема обработки информации при решении задачи распознавания представлена на рис. 1.

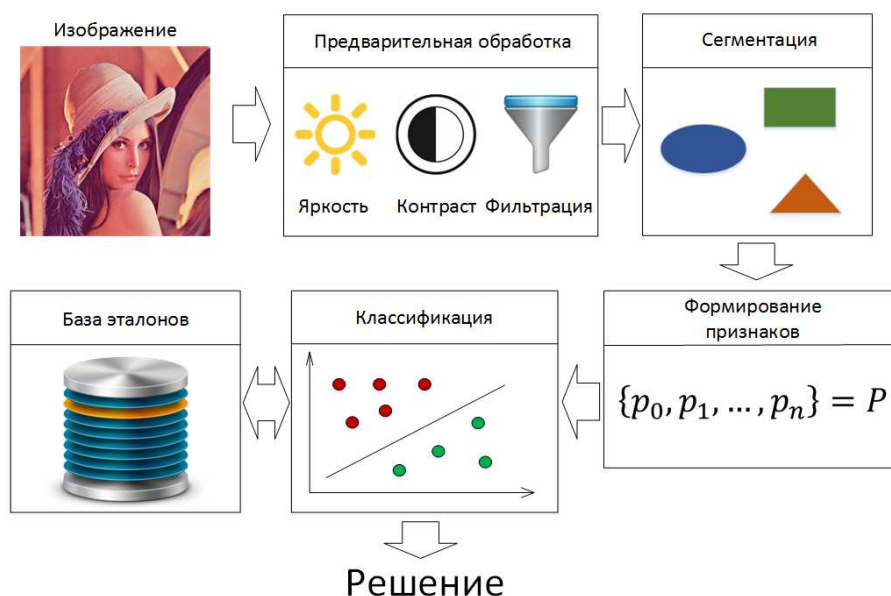


Рис. 1 Стандартная схема обработки информации при решении задачи распознавания

На данный момент для этапов сегментации и формирования признаков не существует строгого теоретического решения, так как в условиях наличия помех эти задачи являются некорректными. И поэтому качественное решение задачи распознавания изображений на указанных этапах не представляется возможным, особенно в режиме реального масштаба времени [1].

Ярким примером этой ситуации является отсутствие надежных алгоритмов для решения практических задач, таких как:

- идентификация личности по фотографии;
- анализ видеопотоков в системах мониторинга для таможни, охраны и т. п.;
- анализ рукописных текстов и (даже) текстов с гостированными шрифтами.

Рассмотрим авторский метод, реализующий новый подход к задаче распознавания образов. Новизна заключается в прямом получении ключевых параметров исследуемого изображения через последовательное применение преобразования Радона и преобразования Фурье.

2 Предлагаемый метод

2.1 Математический аппарат

В основе метода лежат известные преобразования Радона и Фурье.

Рассмотрим преобразование Радона функции двух переменных, так как именно этот случай используется в предлагаемом алгоритме распознавания.

Пусть $f(x, y)$ — функция двух действительных переменных, определенная на всей плоскости и достаточно быстро убывающая на бесконечности (так, чтобы соответствующие несобственные интегралы сходились). Тогда преобразованием Радона функции $f(x, y)$ называется функция

$$R(s, \alpha) = \int_{-\infty}^{\infty} f(s \cos \alpha - z \sin \alpha, s \sin \alpha + z \cos \alpha) dz.$$

Геометрический смысл преобразования Радона — это интеграл от функции вдоль прямой, перпендикулярной вектору $\mathbf{n} = (\cos \alpha, \sin \alpha)$ и проходящей на расстоянии s (измеренном вдоль вектора \mathbf{n} , с соответствующим знаком) от начала координат (рис. 2) [2].

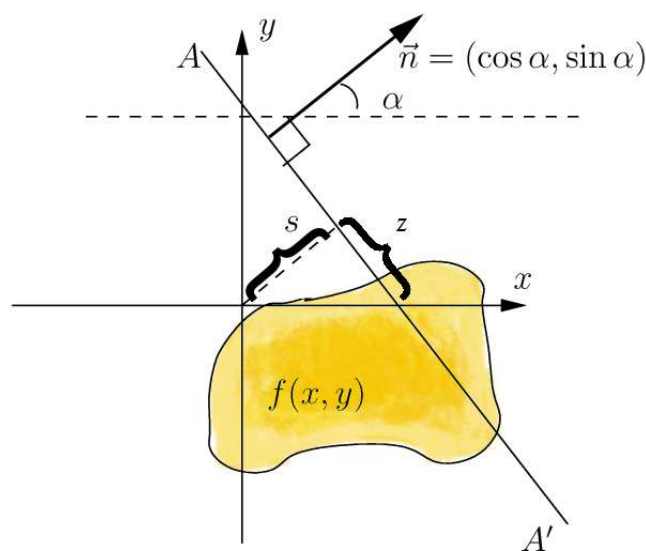


Рис. 2 Двумерное преобразование Радона. В данном случае $R(s, \alpha)$ есть интеграл от $f(x, y)$ вдоль прямой AA'

Напомним, что тригонометрическим рядом Фурье функции $f \in L_2$ с периодом T называют функциональный ряд вида

$$f(x) = \frac{a_0}{2} + \sum_{k=1}^{\infty} \left(a_k \cos \frac{k2\pi x}{T} + b_k \sin \frac{k2\pi x}{T} \right).$$

Синусы и косинусы рядов Фурье можно рассматривать как математические резонаторы, обладающие возможностью обнаруживать присутствие определенных частот в сигнале и при наличии таковых — получать количественную оценку их индивидуального вклада в общий смешанный сигнал. Суммы считаются, как интегралы от $f(x) \cos kx dx$ и $f(x) \sin kx dx$ в границах периода T .

Деление интегралов на T дает нормализованные резонансные отклики, так называемые Фурье коэффициенты — два отклика a_k и b_k для каждого резонансного теста с частотой k :

$$a_k = \frac{1}{T} \int_{x=T_0}^{T_0+T} f(x) \cos \left(kx \frac{2\pi}{T} \right) dx; \quad b_k = \frac{1}{T} \int_{x=T_0}^{T_0+T} f(x) \sin \left(kx \frac{2\pi}{T} \right) dx.$$

Расчет значений a_k и b_k из дискретного потока данных следует правилам [3]:

Дан сигнал $f(x)$ в форме массива длиной N ; N представляет период T , и каждый резонатор записывается в виде массива длиной N . Суммы от 0 до $(N-1)$ заменяют интегралы от T_0 до $(T_0 + T)$:

$$a_k = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \cos \left(kx \frac{2\pi}{N} \right); \quad b_k = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \sin \left(kx \frac{2\pi}{N} \right).$$

2.2 Общий алгоритм

Для начала рассмотрим общий алгоритм, лежащий в основе данного метода. Его можно условно разбить на четыре этапа:

- 1) сбор входных данных. На вход подается изображение-источник, в нем задается искомая область, таким образом получается изображение-образец. Также на вход подается изображение, в котором будет осуществлен поиск данного образца — изображение-тест. Задаются значения следующих параметров:
 - а) количество *интегральных проекций Радона* (ИПР);
 - б) ширина спектра Фурье-резонаторов для каждой ИПР;
 - в) количество первых вычисляемых *Фурье-коэффициентов* (ФК) изображений-теста и изображений-образца (можно воспринимать этот параметр как количество учитываемых при сравнении ФК, однако фактически ненужные ФК не вычисляются с целью экономии ресурсов компьютера);
 - г) допустимое отклонение каждого ФК тестируемого изображения от соответствующего ФК образца;
 - д) допустимое количество существенно отличающихся ФК в полном наборе данных для каждого тестируемого изображения от соответствующих ФК образца;
- 2) построение схемы для вычисления ИПР и массива матриц Фурье-резонаторов согласно заданным на первом этапе параметрам (размерам области образца и значений параметров а и б). Эти схема и массив матриц будут одинаковыми для всех соразмерных изображений и впоследствии будут применяться при вычислении конкретных значений для каждого обрабатываемого изображения;

3) обработка данных. С помощью построенной схемы получают ИПР для изображения-образца, затем с использованием массива матриц резонаторов (и с учетом параметра v) вычисляются ФК для каждой ИПР. Таким образом, получаем массив массивов ФК для изображения-образца. Подобный набор данных для каждого изображения будем называть *проекционными Фурье-данными* (ПФД).

Затем осуществляется проход по всему изображению-тесту со смещением в один пиксель (способы обхода изображения-теста могут варьироваться в зависимости от конкретной задачи). Выделяя таким образом все области соответствующего (изображению-образцу) размера, вычисляются ПФД для каждой области. Посредством специального алгоритма (использующего параметры γ и d) они сравниваются с ПФД изображения-образца, и если тестируемая область считается достаточно похожей на изображение-образец, то параметры этой области дописываются в специальный массив результатов;

4) вывод результатов. В соответствующей вкладке графического интерфейса программы выводится изображение, в котором на белом фоне помещен наиболее близкий к изображению-образцу фрагмент из изображения-теста.

Таким образом, с помощью этого алгоритма можно судить о схожести двух изображений или, например, искать некоторый объект в большом изображении или видеопотоке. Можно регулировать «грубость» работы алгоритма, изменяя входные параметры и, следовательно, детальность учитываемых свойств искомого объекта.

Вычисленный результат работы описанного алгоритма практически аналогичен результату двумерного преобразования Фурье, так как последовательное применение преобразования Радона и одномерного преобразования Фурье эквивалентны [4]. Однако описанный метод требует для обработки матрицы изображения примерно в 2 раза меньше вычислений, чем двумерное преобразование Фурье [5].

2.3 Получение интегральных проекций изображения

В описании общего алгоритма говорилось, что для построения ИПР используется схема, с помощью которой будут вычисляться конкретные значения в ИПР всех соразмерных изображений. Подробно рассмотрим алгоритм построения этой схемы. Для этого необходимы лишь три параметра: ширина и высота изображения и количество выстраиваемых проекций.

Для построения всех ИПР достаточно равномерно пройти по всем (согласно введенному параметру) углам в диапазоне $[-\pi, \pi]$ [4]. С этой целью вычисляется величина шага угла по формуле: $dAngle = \pi / projAmt$, где $projAmt$ — количество выстраиваемых проекций. Затем для каждого номера проекции вычисляется значение соответствующего ему угла ($currAngle = dAngle \cdot projNumb - \pi/2$, где $projNumb$ — номер текущей проекции) и вызывается функция, строящая схему проекции для текущего угла.

Эта функция получает на вход ширину и высоту матрицы изображения и угол направления построения проекции. В основе механизма «сбора» точек матрицы в соответствующий элемент (являющийся массивом) схемы проекции изображения лежит принцип линейной интерполяции в виде модифицированного алгоритма Брезенхема: в элемент схемы проекции добавляются координаты элементов матрицы изображения, ближе всего интерполирующие ломаной прямую, проходящую через элементы этой матрицы. Рассмотрим подробно механизм построения такой схемы.

Напомним, что все возможные значения текущего угла ($currAngle$) могут быть только в диапазоне $[-\pi, \pi]$. Разобьем этот диапазон на четыре сектора: $[-\pi/2, -\pi/4)$, $[-\pi/4, 0)$,

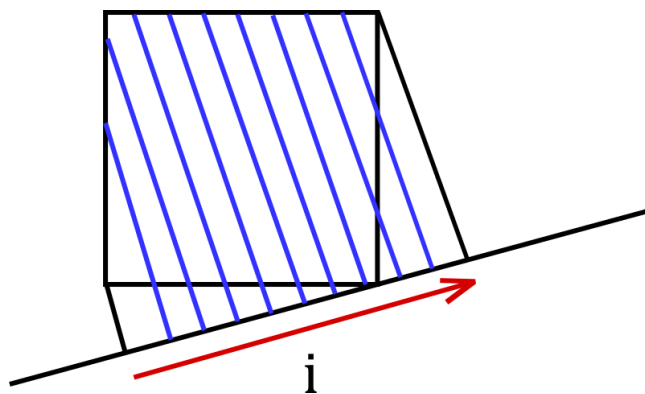


Рис. 3 Линии направлений построения проекции

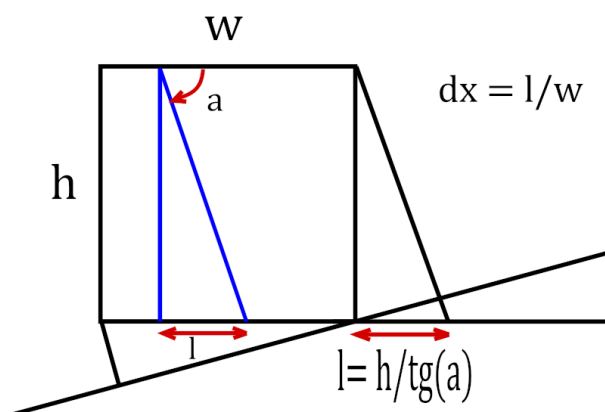


Рис. 4 Вычисление добавочной ширины и величины шага по x

$[0, \pi/4)$, $[\pi/4, \pi/2)$. Не умаляя общности, можно рассмотреть алгоритм построения схемы проекции для угла из первого сектора ($[-\pi/2, -\pi/4)$). Будем проходить по всем значениям i , для которых вдоль линии направления построения проекции будут содержаться элементы матрицы изображения (рис. 3).

Для этого сначала вычислим добавочную ширину $l = h / \text{tg}(\alpha)$, где h — высота матрицы; α — угол, задающий направление построения проекции (он же = currAngle). Затем вычислим величину шага по x : $dx = l/w$, где w — ширина матрицы (рис. 4).

Для каждого значения i (с шагом 1) из достроенного отрезка $[0, w + l]$, перебирая все значения y из отрезка $[0, h]$, будем вычислять x -координату элемента матрицы, лежащего на соответствующей линии направления проекции, по формуле $x = i + \lfloor y dx \rfloor$ (где $\lfloor a \rfloor$ — «пол» числа a). Затем производится проверка выхода получившейся координаты за границы матрицы ($0 \leq y \leq h$, $0 \leq x \leq w$), и, если эта проверка пройдена успешно, получившаяся координата дописывается в i -й элемент проекции (являющийся массивом). Этот процесс отражен на рис. 5.

Общую формулу для сектора $[-\pi/2, -\pi/4)$ можно записать так:

$$P(i) = \{(x, y) : \forall y \in \mathbb{Z} \cap [0, h], x = i + \lfloor y dx \rfloor, 0 \leq x \leq w\}.$$

Для других секторов принцип построения схемы проекции будет тот же, с точностью до знаков и обозначений x и y .

Таким образом, координатам матрицы изображения ставятся в соответствие номера элементов проекции. Это соответствие и позволяет построить схему проекции изображе-

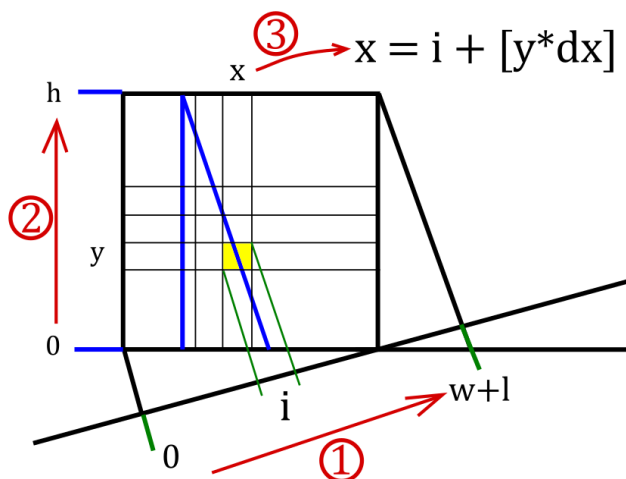


Рис. 5 Механизм отображения элемента матрицы в соответствующую ячейку проекции

ния. Отметим, что направление обхода элементов проекции всегда будет производиться слева направо.

После того как описанная функция построения схемы проекции выполнится для каждого необходимого угла, схема будет построена и записана в память компьютера. В дальнейшем программа будет использовать эту схему для вычисления ИПР любого изображения подходящего размера: сначала один раз для изображения-образца, потом, в среднем случае, значительно большее количество раз для каждой области-теста из изображения-теста, причем вычисляться массивы ИПР будут за один «проход» по матрице обрабатываемого изображения. Заметим также, что на практике элементами каждой ИПР будут являться трехкомпонентные точки (RGB) и суммирование вдоль линии направления угла построения проекции также будет производиться покомпонентно.

Необходимость использования описанной схемы обусловлена тем фактом, что выборка из памяти по одномерному индексу выполняется быстрее, чем целочисленный расчет методом Брезенхема. Благодаря использованию схемы, производительность на этом этапе вычислений увеличилась практически вдвое для случая $80 \times 80 = 6400$ областей-тестов размером $20 \times 20 = 400$ пикселей и примерно в полтора раза для случая $260 \times 260 = 67600$ областей-тестов размером $40 \times 40 = 1600$ пикселей.

2.4 Получение Фурье-коэффициентов для интегральных проекций

Рассмотрим алгоритм построения массива матриц Фурье-резонаторов для соразмерных изображений. Для этого необходимо знать не только размеры изображения и значение параметра компрессии при Фурье-трансформации, но и длины всех ИПР обрабатываемого изображения, значение которых можно получить из схемы для вычисления ИПР. Поэтому построение массива матриц Фурье-резонаторов начинается сразу по завершении построения этой схемы.

Длина массива матриц Фурье-резонаторов будет соответствовать количеству преобразуемых последовательностей, а размеры каждой матрицы резонаторов будут зависеть от длины последовательности и от ширины спектра Фурье-резонаторов. Одно измерение матрицы — это длина последовательности (`projectioni.length`). Количество резонаторов — другое измерение матрицы — будет равно «потолку» от длины последовательности, умноженной на коэффициент ширины спектра:

$$\text{resonatorsAmt}_i = \lceil \text{projection}_i.\text{length} \cdot \text{spectrumWidth} \rceil .$$

Так же, как и в случае со схемой для построения ИПР, массив матриц Фурье-резонаторов записывается в память компьютера и будет использоваться в дальнейшем.

Как говорилось ранее, процесс получения ПФД происходит в два этапа. Сначала с использованием схемы для построения ИПР получают массив проекций, после чего с помощью массива матриц Фурье-резонаторов вычисляются массивы ФК для каждой из таких проекций, таким образом получают ПФД.

На втором этапе дополнительно используется параметр, задающий количество вычисляемых первых ФК для каждой ИПР изображения (`considationPercent`). Осмысленное изменение этого параметра так же, как и в случае параметра компрессии, может способствовать решению проблемы шумоподавления в задаче распознавания образов. Влияние этого параметра, по сути, состоит в том, что при преобразовании последовательности в ФК не сохраняются последние коэффициенты, а ведь именно в последних ФК, отвечающих самым высоким частотам, как правило, содержатся шумы [6].

Таким образом, при вычислении ПФД из массива ИПР изображения достигается эффект низкочастотной Фурье-фильтрации, широко применяемый в задачах радиофизики и теории обработки сигналов [6]. Или же, если для анализа использовать самые первые ФК (т. е. значительно уменьшить значение параметра вычисляемых первых ФК), можно судить о некоторых базовых свойствах изображения.

2.5 Сравнение Фурье-коэффициентов для интегральных проекций двух изображений

Благодаря использованию описанных выше алгоритмов получают ПФД для изображения-образца, затем, «проходя» по всему изображению-тесту, выделяются соразмерные изображению-образцу области и вычисляются ПФД для каждой такой области, после чего производится сравнение ПФД изображения-образца и области-теста.

Напомним, что представляют собой ПФД. Это массив, в котором для каждой ИПР изображения хранится некоторый массив ФК, построенный с учетом заданных параметров.

Идея метода сравнения ПФД двух изображений состоит в том, чтобы подсчитать среднее количество существенно отклоненных ФК из ПФД тестируемого изображения от ФК из ПФД изображения-образца и затем, в зависимости от этой величины, принимать решение об уровне схожести сравниваемых изображений.

Этот метод использует два последних параметра («г» и «д») из списка, упоминавшегося выше.

Первый из них — допустимый процент отклонения каждого ФК тестируемого изображения от соответствующего ФК образца, т. е. если фактическая величина отклонения ФК тестируемого изображения окажется больше значения этого параметра, то этот ФК будет считаться существенно отличающимся от соответствующего ФК образца.

Фактический процент отклонения одного ФК вычисляется как отношение значения разницы ФК тестируемого изображения от соответствующего ФК образца к значению этого ФК образца:

$$\text{aDiv}_c = \frac{|\text{sample}_i.a_x.c - \text{test}_i.a_x.c|}{|\text{sample}_i.a_x.c|}; \quad \text{bDiv}_c = \frac{|\text{sample}_i.b_x.c - \text{test}_i.b_x.c|}{|\text{sample}_i.b_x.c|}.$$

Здесь и далее i — индекс номера проекции, x — индекс номера элемента в массиве ФК, соответствующем одной проекции (ИПР после преобразования Фурье), а c принимает значения из $\{R, G, B\}$ в зависимости от цвета компоненты, для которой вычисляется отклонение.

Затем вычисляется среднее арифметическое для всех отклонений ФК каждой компоненты цвета (RGB) и типа ФК (a и b). Укажем общую формулу для подсчета фактического отклонения одного элемента ПФД:

$$\text{eleDiv}_{ix} = \frac{1}{6} \sum_{c=\{R,G,B\}} (\text{aDiv}_c + \text{bDiv}_c),$$

где aDiv_c и bDiv_c вычислены по формулам, указанным выше, для соответствующих значений i и x .

Далее производится подсчет количества существенно отклоненных ФК в каждой ИПР (strongDivAmt_i) в зависимости от введенного параметра допустимого процента отклонения. После этого вычисляется доля существенно отклоненных элементов массива ФК от их общего количества:

$$\text{strongDivAmtAvg}_i = \frac{\text{strongDivAmt}_i}{\text{sample}_i.\text{length}},$$

и затем считается доля существенно отклоненных ФК во всех ПФД для сравниваемого изображения:

$$\text{strongDivAmtAvgSummary} = \frac{1}{\text{projAmt}} \sum_{i=0}^{\text{projAmt}-1} \text{strongDivAmtAvg}_i.$$

Это значение будем называть отклонением (от образца) ПФД для изображения.

После получения этого значения нужно принять решение о том, считать ли тестируемое изображение достаточно похожим на образец. И в этот момент играет роль второй из управляющих параметров метода сравнения — допустимое количество существенно отличающихся ФК в ПФД для изображения. С ним сравнивается полученное для каждого изображения отклонение ПФД, и если оно меньше заданного параметра, то тестируемое изображение считается достаточно похожим на изображение-образец и информация о нем добавляется в специальный массив результатов поиска. В эту информацию входят координаты найденного похожего изображения и значение отклонения его ПФД.

По завершении процесса поиска схожих изображений массив результатов сортируется по неубыванию значения отклонения ПФД. За счет этого наиболее похожая на образец область оказывается на первой позиции в массиве результатов. Используя информацию о положении этой области, программа выводит соответствующую часть изображения-теста в качестве результата поиска образа из изображения-источника.

Суммарную асимптотическую вычислительную сложность алгоритма можно выразить формулой:

$$f(n, m, x, y, p, s, c) = \Theta \left((m - x)(n - y)(x^2 + y^2) \left(p + 12(x^2 + y^2) \frac{s c}{p} \right) \sum_{k=0}^{p/4} \text{tg} \left(k \frac{\pi}{p} \right) \right),$$

где n, m — размеры изображения-теста; x, y — размеры изображения-образца; p — количество ИПР (projAmt); s — ширина спектра Фурье-резонаторов (spectrumWidth); c — количество первых вычисляемых ФК ($\text{considerationPercent}$).

3 Результаты

Рассмотрим результаты, полученные с помощью программного прототипа, реализующего описанный метод распознавания образов. Как отмечалось ранее, очень большое

значение имеют параметры, задаваемые перед запуском поиска. Именно от них зависит качественная эффективность работы алгоритма. Оптимальный выбор набора параметров зависит от конкретной задачи. Ниже приведены примеры таких задач.

Для оценок ошибок будет дополнительно использоваться модификация программного прототипа, в качестве результата отображающая не только наиболее похожее на образец изображение, но и все изображения, сочтенные схожими (содержащиеся в массиве с найденными областями). На основе полученных результатов будут делаться выводы о том, какие значения управляющих параметров следует устанавливать в зависимости от конкретного изображения, выбранного в качестве образца.

3.1 Простые образы

Для начала исследуем возможности алгоритма к распознаванию простейших геометрических форм (рис. 6) и изучим особенности решения этой задачи.



Рис. 6 Простейшие геометрические формы

В качестве изображения-теста будет использоваться то же изображение с добавленными шумами (рис. 7).

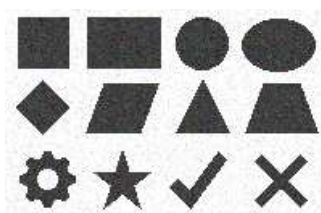


Рис. 7 Зашумленное изображение простейших геометрических форм

Начнем, например, с поиска квадрата. Образцом выбран серый квадрат (без белой границы). В подписях к рисункам с результатами поисков будут указываться использованные управляющие параметры в последовательности, изложенной в описании общего алгоритма.

Как и ожидалось, в массив результатов попал сам квадрат и прямоугольник (рис. 8). Произошло это потому, что прямоугольник содержит в себе искомый образ (квадрат), и, соответственно, он успешно распознается и попадает в массив результатов. Причем этот эффект не является ошибкой второго рода.

Теперь будем искать шестеренку. Поскольку ее форма достаточно необычна, то при правильном выборе управляющих параметров поиска найдется только она сама (рис. 9).

Однако заметим, что, как говорилось ранее, при неверном задании параметров могут возникать ошибки. Так, в том случае, если требовать слишком строгого сходства, не будет найдено вообще ни одного совпадения (на выходе будет получено пустое изображение —

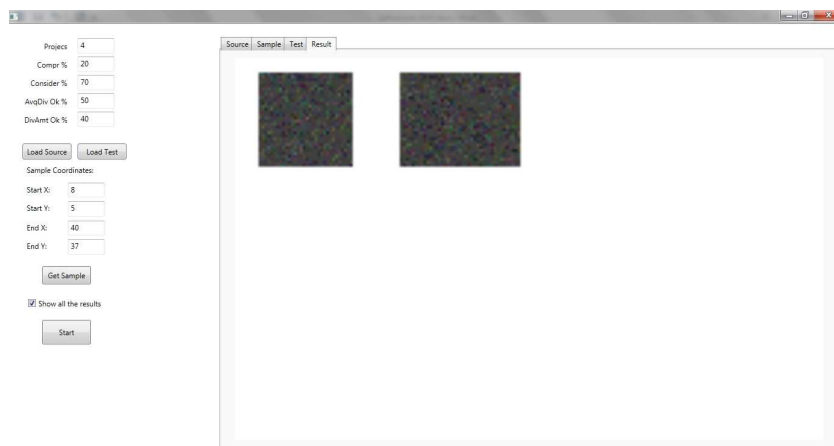


Рис. 8 Результат поиска квадрата (4, 20, 70, 50, 40)



Рис. 9 Результат поиска шестеренки (4, 20, 70, 30, 40)

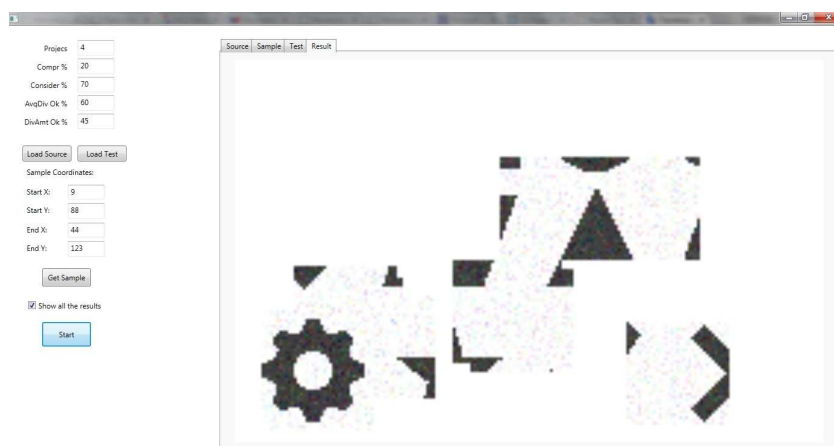


Рис. 10 Ошибки второго рода при поиске шестеренки (4, 20, 70, 60, 45)

белый фон) — ошибка первого рода. В то же время, если позволить считать успешным распознаванием слишком слабое совпадение, то в результат войдут области, не имеющие практически ничего общего с образцом (рис. 10) — ошибка второго рода.

Ранее отмечалось, что управляющие параметры следует подбирать исходя из желаемой «грубости» поиска. Приведем некоторые формулы, позволяющие выбрать значения этих параметров в условиях конкретной задачи.

В зависимости от параметра ширины спектра при переходе от ИПР к ее ФК можно менять количество Фурье-резонаторов, используемых при преобразовании. Их число можно получить по формуле $\text{spectrumWidth} \cdot \text{projection.length}$. Параметр вычисляемых первых ФК для ИПР следует умножить на это число, тогда получим количество ФК для каждой ИПР в ПФД изображения: $\text{considerationPercent} \cdot \text{spectrumWidth} \cdot \text{projection.length}$.

Общая формула для минимального размера учитываемого элемента изображения будет выглядеть так: $\min \{ \text{img.height}, \text{img.width} \} / (\text{considerationPercent} \cdot \text{spectrumWidth})$.

3.2 Дорожные знаки

Теперь отойдем от тестовых изображений и покажем эффективность работы алгоритма при решении реальных задач.

Например, существует задача распознавания дорожных знаков, качественного решения которой на данный момент не существует. Описанный алгоритм может стать таким решением. Возьмем в качестве образца изображение знака «пешеходный переход», четкое и ровное (рис. 11).



Рис. 11 Изображение знака «пешеходный переход»

В качестве изображения теста возьмем фотографию с реальным дорожным знаком (рис. 12).



Рис. 12 Реальный дорожный знак

Выделим в этом изображении сам квадрат знака (рис. 13) (например, с помощью примитивов Хаара [7]) и попробуем распознать в нем образец.

При точном выборе управляющих параметров поиска удается распознать на фотографии «образцовый» дорожный знак (рис. 14).



Рис. 13 Выделенный из фотографии квадрат знака



Рис. 14 Результат распознавания знака (10, 7, 75, 70, 70)

Отметим, что в результатах отображаются все найденные совпадения, и они являются очень точными. Это значит, что удалось избежать ошибок второго рода. Также удалось избежать и ошибок первого рода: можно видеть, что знак распознан верно, несмотря на дефект при печати самого знака (горизонтальная белая линия в центре) и надпись внизу знака, искажающую его часть.

3.3 Отслеживание глаз (айтрекинг)

Другая практическая задача, нуждающаяся в решении, — качественное распознавание зрачка человека. Существующие методы не дают высокой точности, и в результате при попытке отслеживать движения зрачка в видеопотоке происходит «дрожание» распознанного зрачка, даже если в реальности зрачок неподвижен. Рассматриваемый в данной статье алгоритм распознавания может решить эту проблему. Продемонстрируем это на примере. Единоразы указав в качестве образца зрачок с частью радужной оболочки (рис. 15), можно отслеживать его положение в кадре.

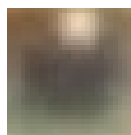


Рис. 15 Изображение зрачка глаза человека и части радужной оболочки

Теперь попробуем найти зрачки в двух разных кадрах видео (рис. 16).

Выделим в этих кадрах области глаз (рис. 17).

После этого запустим поиск зрачков в полученных изображениях. Очевидно, что программа справляется с этой задачей, причем с минимальными ошибками второго рода, которые, в свою очередь, можно устранить выбором наиболее близкой к образцу области в районе каждого глаза. Снимки экрана с результатами представлены на рис. 18 и 19.

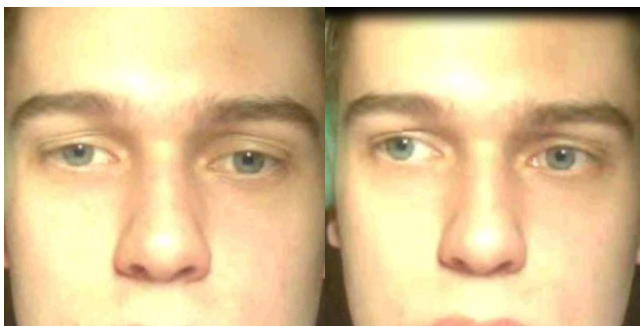


Рис. 16 Два различных кадра из видеозаписи с лицом человека



Рис. 17 Выделенные из кадров фрагменты с изображениями глаз человека

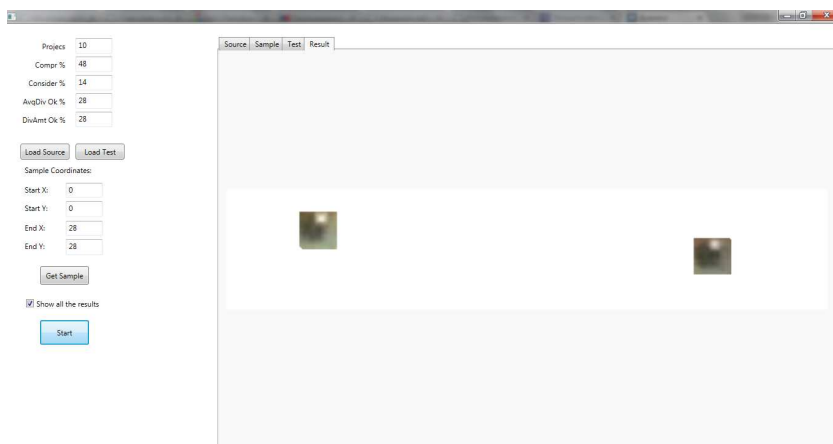


Рис. 18 Результат поиска зрачков в фрагменте из первого кадра (10, 48, 14, 28, 28)

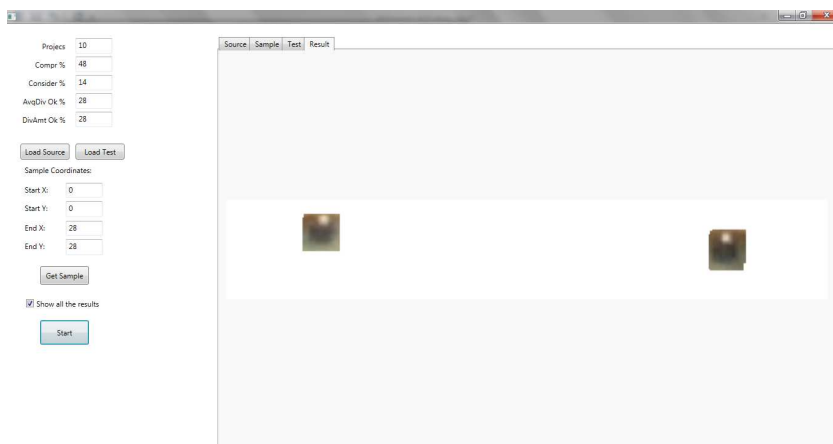


Рис. 19 Результат поиска зрачков в фрагменте из второго кадра (10, 48, 14, 28, 28)

3.4 Перспективы

За счет универсальности подхода при анализе изображения рассмотренный метод может применяться в широком спектре задач, возникающих в области обработки данных.

В современном мире в условиях постоянно растущего объема передаваемых по различным каналам данных задача их обработки становится одной из наиболее актуальных [8]. Описанный метод позволяет приблизиться к решению этой задачи в случае, когда в качестве передаваемых данных служит информация о свойствах изображения и его содержании. Анализируя ПФД некоторого изображения, можно получить эту информацию, причем объем данных, занимаемых ПФД, значительно меньше, чем объем сильно сжатого (даже с потерей данных) изображения.

Другая сфера применения метода — контроль качества графических данных. Как упоминалось ранее, используемый алгоритм способен бороться с шумами и помехами в изображениях. Этот механизм также позволяет анализировать количество и интенсивность шумов и помех, позволяя тем самым делать выводы о качестве изображений или видеопотоков.

Еще одна задача обработки данных, одна из самых сложных и вместе с тем актуальных, — задача распознавания. Она относится к области «машинного зрения» (Computer Vision, или CV). На сегодняшний день существует множество различных методов анализа объектов в изображении, однако все они являются сильно специфическими и приспособлены для решения задач в узких областях [9]. Представленный метод отличается универсальностью, т. е. он может применяться для решения задач распознавания любых объектов с любой заранее выбранной глубиной точности. Также этот метод может применяться при отслеживании положения объектов в кадре, т. е. в задаче ведения распознанного объекта в видеопотоке. Такая необходимость возникает, например, в системах автоматического мониторинга видеонаблюдения, мониторинга контента телевизионных каналов, отслеживания движений человека. В частности, метод позволяет с высокой точностью отслеживать движения зрачков человека, что находит применение в компьютерной окулографии.

Для повышения толерантности алгоритма к поворотам сравниваемых матриц изображений относительно друг друга необходимо использовать дополнительный механизм определения начальной проекции для сравнения. При этом достаточно использовать только первые несколько ФК для подряд идущих проекций. Это позволит добиться распознавания искомого объекта с произвольным углом поворота при незначительном увеличении вычислительной нагрузки.

Чтобы справиться с задачей распознавания, когда разрешение изображения искомого объекта отличается от разрешения изображения-образца, достаточно добиться синхронизации количества результирующих ФК в ПФД тестируемого изображения и количества ФК в ПФД изображения-образца. Для этого необходимо умножить значение исходной ширины спектра на коэффициент $k = \text{size}_{\text{sample}}/\text{size}_{\text{test}}$ отношения размера изображения-образца к размеру области-теста $\text{spectrumWidth}_{\text{test}} = k \cdot \text{spectrumWidth}_{\text{sample}}$ и производить преобразование Фурье вдоль проекции изображения-теста с шагом, кратным тому же коэффициенту k . Совпадение значений ФК для одного объекта на изображениях разных размеров наглядно видно на графиках, изображенных на рис. 20.

В перспективе с использованием механизмов «машинного обучения» (Machine Learning, или ML) описанный метод может применяться для создания системы интеллектуального анализа графических данных. При этом масштабы задействованных вычислительных мощностей и объемы передаваемых данных будут несравнимо меньше, чем у подобных систем, основанных на искусственных нейронных сетях.

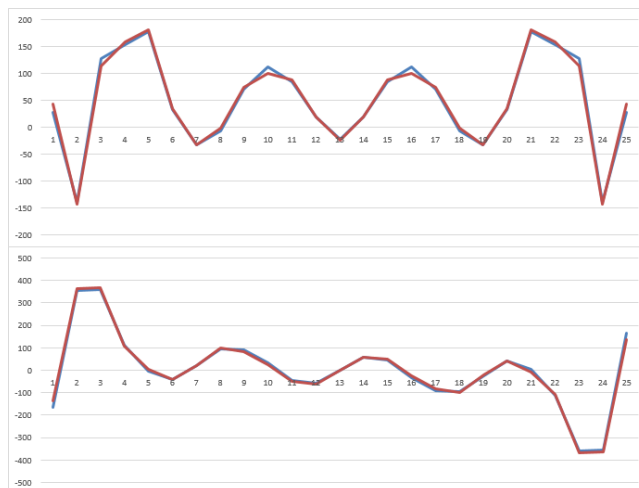


Рис. 20 Графики значений ФК для изображений разных размеров

4 Заключение

Представленный в данной статье метод, основанный на комбинации классических преобразований Радона и Фурье, а также статистическом анализе, реализован в виде рабочего алгоритма, успешно решающего поставленную задачу мониторинга видеопотока и анализа изображений. Метод уже внедрен для решения некоторых задач, в частности в системе компьютерной окулографии, используемой для диагностики нейропатологий через отслеживание движений глаз (айтрекинг).

Получены данные о сильных и слабых сторонах алгоритма и возможных методах оптимизации исполняемого кода программного прототипа.

С учетом свойств данных, полученных при применении метода к конкретному изображению, были сделаны выводы о возможных областях применения алгоритма и перспективах его использования.

Литература

- [1] Новая информационная технология обработки произвольных изображений — ТАРе-технология. <http://comexp.ru/node/129>.
- [2] Грузман И. С. Математические задачи компьютерной томографии // Соросовский образовательный журнал, 2001. № 5.
- [3] Зорич В. А. Математический анализ. — М.: Физматлит, 1984. 544 с.
- [4] Тихонов А. Н., Арсенин В. Я., Тимонов А. А. Математические задачи компьютерной томографии. Серия: Проблемы науки и технического прогресса. — М.: Наука, 1987. 160 с.
- [5] Gertner I. New efficient algorithm to complete the two-dimensional discrete Fourier transform // IEEE Trans. ASSP, 1988. Vol. 36. No. 7. P. 1036–1050.
- [6] Сергиенко А. Б. Цифровая обработка сигналов. — 2-е изд. — СПб.: Питер, 2006. 751 с.
- [7] Lienhart R., Maydt J. An extended set of Haar-like features for rapid object detection // ICIP02, 2002. P. 900–903.
- [8] Горелик А. Л., Скрипкин В. А. Методы распознавания. — 4-е изд. — М.: Высшая школа, 2004. 262 с.
- [9] Фомин Я. А. Распознавание образов: теория и применения. — 2-е изд. — М.: ФАЗИС, 2012. 429 с.

Поступила в редакцию 23.07.2015

References

- [1] Novaya informatsionnaya tekhnologiya obrabotki proizvol'nykh izobrazheniy — TAPe-tekhnologiya. [New information technology of processing arbitrary images — TAPe-technology]. Available at: <http://comexp.ru/node/129> (accessed December 11, 2015). (In Russian.)
- [2] Gruzman, I. S. 2001. Matematicheskie zadachi komp'yuternoy tomografii [Mathematical problems of computed tomography]. *Sorosovskiy Obrazovatel'nyy Zhurnal* 5. (In Russian.)
- [3] Zorich, V. A. 1984. *Matematicheskiy analiz* [Mathematical analysis]. Moscow: Fizmatlit. 544 p. (In Russian.)
- [4] Tikhonov, A. N., V. Ya. Arsenin, and A. A. Timonov. 1987. Matematicheskie zadachi komp'yuternoy tomografii [Mathematical problems of computed tomography]. Ser. Problemy nauki i tekhnicheskogo progressa [Problems of science and technological progress ser.] Moscow: Nauka. 160 p. (In Russian.)
- [5] Gertner, I. 1988. New efficient algorithm to complete the two-dimensional discrete Fourier transform. *IEEE Trans. ASSP* 36(7):1036–1050.
- [6] Sergienko, A. B. 2006. *Tsifrovaya obrabotka signalov* [Digital signal processing]. 2nd ed. St. Petersburg: Piter. 751 p. (In Russian.)
- [7] Lienhart, R., and J. Maydt. 2002. An extended set of Haar-like features for rapid object detection. *ICIP02*. 900–903.
- [8] Gorelik, A. L., and V. A. Skripkin. 2004. *Metody raspoznavaniya* [Methods of recognition]. 4th ed. Moscow: Vysshaya Shkola. 262 p. (In Russian.)
- [9] Fomin, Ya. A. 2012. *Raspoznavanie obrazov: Teoriya i primeneniya* [Pattern recognition: Theory and application]. 2nd ed. Moscow: FAZIS. 429 p. (In Russian.)

Received July 23, 2015