

УДК 004.932

А.Н. Козловский

ОПРЕДЕЛЕНИЕ УГЛОВЫХ ТОЧЕК НА ОСНОВЕ АППРОКСИМАЦИИ КОНТУРОВ ОБЪЕКТОВ ИЗОБРАЖЕНИЯ

Рассматривается детектор угловых точек на основе аппроксимации контуров объектов изображения. В основе работы детектора лежат разработанные схемы определения отсчета бинарного изображения «претендентом» на угловую точку; алгоритм построения описания бинарного контура объекта изображения для отсчета-«претендента» на угловую точку в терминах цепного кода Фримана; правила идентификации угловых точек на основе аппроксимации полученного описания бинарного контура. Тестирование предложенного детектора подтвердило его приемлемую эффективность при нахождении угловых точек.

Введение

Задача нахождения угловых точек объектов изображения является важной компонентой в различных приложениях цифровой обработки изображений. Например, она используется для определения опорных точек в задаче взаимного совмещения изображений, а также находит широкое применение в задачах создания компьютерного зрения. Этим и объясняется большое количество известных в настоящее время детекторов угловых точек.

Детекторы Моравека и Харриса [1, 2] определяют угловые точки в режиме скользящего окна, т. е. не по всему изображению, а только по ограниченной области – окну обработки при сдвиге его по изображению, и используют различные критерии для определения принадлежности отсчетов-«претендентов» к классу угловых. Данные алгоритмы инвариантны по отношению к повороту и нечувствительны к яркости изображения, но недостаточно точно определяют угловые точки и чувствительны к шуму.

Для решения проблем ранее известных детекторов угловых точек был разработан детектор CSS [3] – многомасштабный алгоритм, основанный на использовании масштаба пространства кривизны, который может определять углы на плоских кривых.

Детектору CSS присущи следующие недостатки: использование фильтров оценки частных производных; определение «претендентов» на угловые точки через локальные экстремумы функции кривизны развивающейся кривой ($K(u, \delta)$), которые, в свою очередь, могут быть угловыми точками, ложными угловыми точками, шумом; нахождение ложных угловых точек на контуре, имеющем круговую форму, а также при наличии в некоторой небольшой окрестности изображения нескольких «претендентов» на угловую точку.

Улучшенный детектор CSS [4] решает проблемы детектора CSS за счет использования различных значений масштаба дескриптора CSS для контуров с различной длиной, а также путем сглаживания функции контура объекта, имеющего достаточно большую длину, для удаления ложных максимумов. Однако критерий выбора длин контуров недостаточно ясен, как и то, что значение параметра масштаба должно зависеть от длины рассматриваемого контура.

Х.С. Не и N.H.C. Yung [5] предложили CSS-подобный детектор, который точнее определяет угловые точки, чем улучшенный детектор CSS, и имеет следующие особенности: вычисление модуля $K(u, \delta)$ с использованием относительно малого масштаба; применение локального адаптивного порога для отсеивания ложных угловых точек на контуре, имеющем круговую форму; устранение ложных угловых точек, возникающих в результате шумов или локальных особенностей бинарного контура объекта, путем оценки отсчета-«претендента» в некоторой локальной области.

В настоящей работе предлагается детектор угловых точек, который находит угловые точки, соответствующие углам, образованным между направленными отрезками. Нахождение осуществляется на основе проверки возможности аппроксимации бинарных контуров объектов изображения, исходящих из угол образующих отсчетов и описанных в терминах цепного кода Фримана, отрезками прямых с заданной точностью.

1. Детектор угловых точек на основе аппроксимации контуров объектов изображения

Пусть имеется изображение, содержащее некоторую совокупность объектов на однородном фоне. Требуется определить угловые точки объектов изображения и их координаты. Без потери общности можно считать, что изображение бинарное, т. е. значения отсчетов, соответствующих объектам, равны единице, а отсчеты фона имеют нулевые значения. Полутоновое изображение всегда может быть приведено к бинарному в результате пороговой обработки.

Следует отметить, что задача нахождения угловых точек объектов изображения и определения их координат относится к классу задач измерений на изображениях. Суть задач данного класса состоит в получении по имеющимся данным количественных значений параметров, характеризующих изображение в целом или отдельные его объекты. Анализ поставленной задачи показывает, что угловые точки объектов изображения могут быть определены с помощью алгоритма обработки бинарного изображения. При использовании этого алгоритма все искомые параметры определяются за один проход по изображению, например при его построчной развертке [6].

Угол есть фигура (рис. 1), образованная двумя лучами OA и OB (стороны угла), исходящими из одной точки O (вершина угла) [7]. Таким образом, если на лучах OA и OB от точки O отложить отрезки OM и ON , то рассматриваемый детектор угловых точек позволяет находить вершины углов – угловые точки, которые принадлежат углам, образованным между направленными отрезками \overline{OM} и \overline{ON} (см. рис. 1). Нахождение осуществляется путем определения отсчета изображения как «претендента» на угловую точку (точка O) согласно разработанным схемам (рис. 2), дальнейшего описания бинарного контура объекта изображения для отсчета «претендента» в терминах цепного кода Фримана и идентификации угловой точки на основе аппроксимации полученного описания соответствующего бинарного контура векторами \bar{a} и \bar{b} .

Длины векторов \bar{a} и \bar{b} полагаются равными и не должны превышать n отсчетов, где n – один из параметров детектора, определяемый пользователем. В этом случае нахождение угловых точек объектов изображения осуществляется за четыре шага:

- 1) контурная сегментация исходного изображения;
- 2) нахождение отсчета-«претендента» на угловую точку;
- 3) построение описания бинарного контура объекта изображения для отсчета-«претендента»;
- 4) идентификация угловой точки.

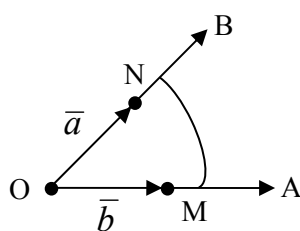


Рис. 1. Угол BOA

На первом шаге выполняется контурная сегментация исходного изображения путем применения детектора краев, например детектора Канни [8]. Могут также применяться дополнительные процедуры: соединение пикселей краев объектов, разделенных промежутком в один пиксел; удаление отростков пикселей. Ширина бинарного контура объектов изображения равняется одному пикселу.

На втором шаге выполняется нахождение отсчета-«претендента» на угловую точку за один проход по исходному изображению при его построчной развертке, т. е. слева направо в строке и сверху вниз по строкам. Рассмотрим произвольный отсчет $f(j, i)$, принадлежащий матрице отсчетов исходного изображения, где j – порядковый номер строки, содержащей отсчет, а i – порядковый номер отсчета в строке. Отсчет может быть отнесен к одному из двух типов отсчетов-

«претендентов» на угловую точку при выполнении ряда условий, схемы которых представлены на рис. 2. Рассматриваемый отсчет обозначен крестиком. Отсчеты изображения, обозначенные черными квадратами, будем называть углом образующими отсчетами, так как равенство их единице является необходимым условием для определения рассматриваемого отсчета изображения «претендентом» на угловую точку.

Отсчет $f(j, i)$ является отсчетом-«претендентом» на угловую точку первого типа в случае равенства угол образующих отсчетов ($f(j, i+1)$ и $f(j+1, i)$) единице (см. рис. 2, а), а также равенства одного из отсчетов $f(j-1, i+2)$ или $f(j+2, i-1)$ нулю (на схеме они обозначены нулями). Отсчеты, обозначенные двойками и тройками, не должны равняться единице одновременно. Слева или справа рассматриваемых угол образующих отсчетов изображения должны отсутствовать подобные угол образующие отсчеты.

Отсчет $f(j, i)$ является отсчетом-«претендентом» на угловую точку второго типа в случае равенства угол образующих отсчетов ($f(j, i-1)$ и $f(j+1, i)$) единице (рис. 2, б), а также равенства одного из отсчетов $f(j-1, i-2)$ или $f(j+2, i+1)$ нулю (на схеме они обозначены нулями). Отсчеты, обозначенные двойками и тройками, не должны равняться единице одновременно. Слева или справа рассматриваемых угол образующих отсчетов изображения должны отсутствовать подобные угол образующие отсчеты.



Рис. 2. Схемы определения отсчета бинарного изображения «претендентом» на угловую точку: а) первого типа; б) второго типа

На третьем шаге выполняется построение описания бинарного контура объекта изображения для отсчета-«претендента» в терминах 8-связного цепного кода Фримана. Контур для каждого отсчета-«претендента» состоит из двух частей. Первая часть – «верхний» контур (вектор \bar{a} (см. рис. 1)) – описывается, начиная с угол образующего отсчета $f(j, i+1)$ или $f(j, i-1)$ (см. рис. 2). Вторая часть – «нижний» контур (вектор \bar{b} (см. рис. 1)) – описывается, начиная с угол образующего отсчета $f(j+1, i)$ (см. рис. 2). Кодирование направлений при построении описания бинарных контуров осуществляется согласно предложенному способу (рис. 3).

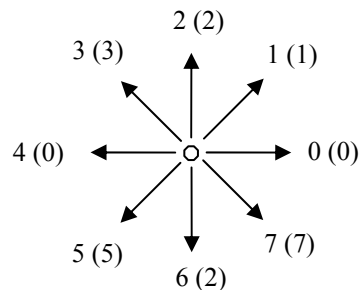


Рис. 3. Нумерация и кодирование направлений 8-связного цепного кода Фримана

На последнем четвертом шаге выполняется идентификация угловой точки посредством разработанных правил идентификации на основе аппроксимации полученного на предыдущем этапе описания бинарного контура объекта изображения для отсчета-«претендента» на угловую точку векторами \bar{a} и \bar{b} .

Рассматриваемый детектор угловых точек использует в своей работе следующие параметры:

n – длина направленных отрезков \overrightarrow{OM} и \overrightarrow{ON} , между которыми определяется угол; измеряется в отчетах бинарного контура объекта изображения;

wk – число ложных допустимых направлений в правилах идентификации;

mx , my – ширина и высота области изображения, в которой не может находиться больше чем одна угловая точка с центром в последней найденной угловой точке;

$threshold$ – порог, используемый при идентификации угловых точек; вычисляется по формуле

$$threshold = floor(2 * n / 3), \quad (1)$$

где $floor$ – операция округления до ближайшего целого в меньшую сторону. Данный параметр используется при идентификации ложных угловых точек. Например, если количество направлений с кодом 1 в описании «верхнего» контура и количество направлений с кодом 5 в описании «нижнего» контура больше (или равно) определенного порога, то отчет-«претендент» на угловую точку, соответствующий контурам с таким описанием, считается ложным.

2. Алгоритм построения описания бинарного контура объекта изображения для отчета-«претендента» на угловую точку в терминах цепного кода Фримана

В настоящее время в литературе известно достаточно большое количество алгоритмов анализа бинарных контуров объектов изображения, обладающих своими достоинствами и недостатками [6, 9]. В настоящей работе рассматривается один из возможных вариантов анализа бинарных контуров объектов изображения на основе описания бинарного контура объекта изображения в терминах 8-связного цепного кода Фримана. Предлагаемый алгоритм обладает следующими особенностями: кодирование направлений выполняется на основе схемы нумерации и кодирования направлений 8-связного цепного кода Фримана (рис. 3), что упрощает анализ описанного контура и увеличивает быстродействие работы детектора; прослеживание контура, т. е. непрерывный переход от текущей точки контура к последующей, достигается за счет запоминания двух последних шагов перехода; выбор направления следующего перехода при прослеживании контура осуществляется согласно назначенным приоритетам для каждого направления осмотра, а также правилам выбора направления перехода, что и позволяет описывать бинарный контур удовлетворительно.

Описание бинарного контура объекта изображения выполняется на основе 8-связного цепного кода Фримана, с помощью которого контур представляется в виде последовательности соединенных отрезков, для которых указаны длина и направление [10]. Примем длину кодируемого отрезка равной единице (один пиксел), тогда каждый отрезок может быть представлен лишь его направлением (переходом), которое кодируется числом в соответствии со схемой нумерации, представленной на рис. 3. Код для каждого из переходов указан в скобках.

Поиск ближайшего соседнего отчета в описываемом бинарном контуре объекта изображения для найденного отчета-«претендента» выполняется согласно одному из двух типов осмотров направлений: осмотр по часовой стрелке – 76543210, осмотр против часовой стрелки – 56701234. Таким образом, построение описания «верхнего» и «нижнего» контуров для отчета-«претендента» осуществляется на основе следующих правил:

– отчет-«претендент», а также отчет, обозначенный на схемах единицей (см. рис. 2), не должны принимать участие в построении описания бинарного контура объекта изображения;

– начальное направление осмотра может быть задано посредством некоторой переменной для каждого из двух типов осмотров направлений;

– алгоритм прослеживания контура не должен допускать перехода назад по контуру, что достигается посредством двух переменных, которым присваиваются значения противоположного направления текущему переходу и переходу, осуществленному на предыдущем шаге;

– для отсчета-«претендента» первого типа построение описания «верхнего» контура выполняется на основе осмотра против часовой стрелки, начиная с направления с номером 7 (0123), а построение описания «нижнего» контура выполняется на основе осмотра по часовой стрелке, начиная с направления с номером 7 (6543);

– для отсчета-«претендента» второго типа построение описания «верхнего» контура выполняется на основе осмотра по часовой стрелке, начиная с направления с номером 5 (4321), а построение описания «нижнего» контура выполняется на основе осмотра против часовой стрелки, начиная с направления с номером 5 (6701);

– для отсчета-«претендента» первого типа в случае равенства единице отсчетов изображения, обозначенных тройками на рис. 2, а, построение описания «верхнего» контура начинается с направления с номером 1, а «нижнего» контура – с направления с номером 5;

– для отсчета-«претендента» первого типа в случае равенства единице отсчетов изображения, обозначенных двойками на рис. 2, а, построение описания «верхнего» контура заканчивается на направлении с номером 1, а «нижнего» контура – на направлении с номером 5;

– для отсчета-«претендента» второго типа в случае равенства единице отсчетов изображения, обозначенных тройками на рис. 2, б, построение описания «верхнего» контура начинается с направления с номером 3, а «нижнего» контура – с направления с номером 7;

– для отсчета-«претендента» второго типа в случае равенства единице отсчетов изображения, обозначенных двойками на рис. 2, б, построение описания «верхнего» контура заканчивается на направлении с номером 3, а «нижнего» контура – на направлении с номером 7;

– приоритет порядка осмотра направлений соответствует порядку осмотра в каждом из двух типов осмотров направлений, т. е. для осмотра по часовой стрелке направление с номером 7 обладает наивысшим приоритетом осмотра, далее по убыванию приоритета следует направление с номером 6 и т. д. При этом направления с номером 0, 2, 4, 6 имеют более высокий приоритет осмотра, нежели направления с номером 1, 3, 5, 7. Например, для осмотра по часовой стрелке переход по направлению с номером 7 выполняется только в том случае, если нельзя пойти по направлению с номером 6;

– переход по направлению с номером 0 выполняется в том случае, если отсчет $f(j, i + 1)$ равен единице и отсчеты $f(j - 1, i + 1)$ и $f(j - 1, i)$ или $f(j + 1, i + 1)$ и $f(j + 1, i)$ равны нулю;

– переход по направлению с номером 1 выполняется в том случае, если отсчет $f(j - 1, i + 1)$ равен единице и отсчет $f(j, i + 1)$ равен нулю для осмотра по часовой стрелке, отсчет $f(j - 1, i)$ равен нулю или один из отсчетов $f(j - 1, i + 1)$, $f(j, i + 1)$ равен единице для осмотра против часовой стрелки;

– переход по направлению с номером 2 выполняется в том случае, если отсчет $f(j - 1, i)$ равен единице и отсчеты $f(j - 1, i - 1)$ и $f(j, i - 1)$ или $f(j - 1, i + 1)$ и $f(j, i + 1)$ равны нулю;

– переход по направлению с номером 3 выполняется в том случае, если отсчет $f(j - 1, i - 1)$ равен единице и отсчет $f(j - 1, i)$ равен нулю или один из отсчетов $f(j - 1, i + 1)$, $f(j, i + 1)$ равен единице для осмотра по часовой стрелке, отсчет $f(j, i - 1)$ равен нулю для осмотра против часовой стрелки;

– переход по направлению с номером 4 выполняется в том случае, если отсчет $f(j, i - 1)$ равен единице и отсчеты $f(j - 1, i - 1)$ и $f(j - 1, i)$ или $f(j + 1, i - 1)$ и $f(j + 1, i)$ равны нулю, но только для осмотра по часовой стрелке;

– переход по направлению с номером 5 выполняется в том случае, если отсчет $f(j + 1, i - 1)$ равен единице и отсчет $f(j, i - 1)$ равен нулю или один из отсчетов $f(j - 1, i - 1)$, $f(j - 1, i)$ равен единице для осмотра по часовой стрелке, отсчет $f(j + 1, i)$ равен нулю или один из отсчетов $f(j + 1, i + 1)$, $f(j, i + 1)$ равен единице для осмотра против часовой стрелки;

– переход по направлению с номером 6 выполняется в том случае, если отсчет $f(j + 1, i)$ равен единице и отсчеты $f(j + 1, i - 1)$ и $f(j, i - 1)$ или $f(j + 1, i + 1)$ и $f(j, i + 1)$ равны нулю;

– переход по направлению с номером 7 выполняется в том случае, если отсчет $f(j+1, i+1)$ равен единице и отсчет $f(j+1, i)$ равен нулю или один из отсчетов $f(j+1, i-1)$, $f(j, i-1)$ равен единице для осмотра по часовой стрелке, отсчет $f(j, i+1)$ равен нулю или один из отсчетов $f(j-1, i+1)$, $f(j-1, i)$ равен единице для осмотра против часовой стрелки;

– построение описания контура прекращается в том случае, если встретились угол образующие отсчеты или были рассмотрены все возможные направления осмотра.

Проанализируем результаты работы представленного алгоритма на примере описания бинарных контуров объектов некоторого изображения в терминах 8-связного цепного кода Фримана (рис. 4). В результате работы алгоритма были получены следующие описания контуров: «верхний» контур – 00070000, «нижний» контур – 70070707 (рис. 4, а); «верхний» контур – 10110121, «нижний» контур – 00300500 (рис. 4, б). Заметим, что на рис. 4 крестиком обозначены угловые точки, а черными квадратами – угол образующие отсчеты.

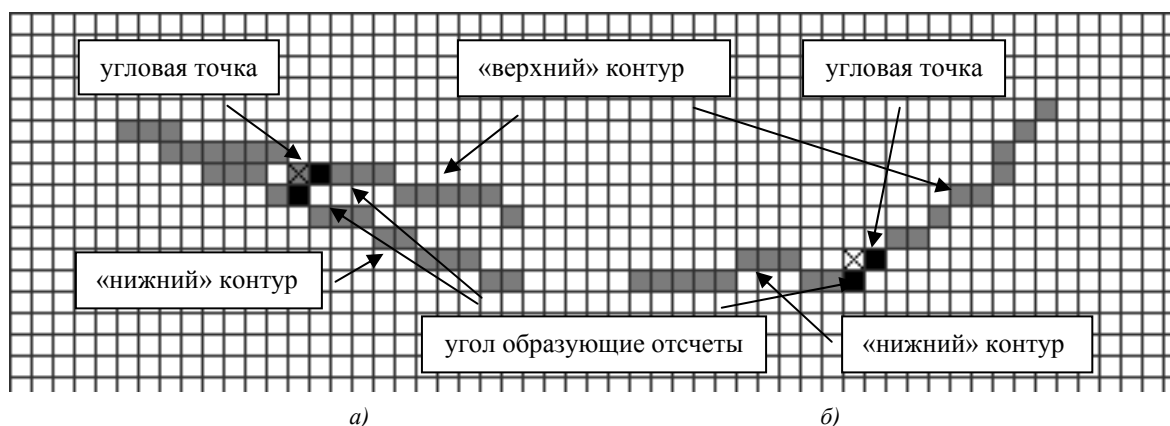


Рис. 4. Тестируемые бинарные контуры объектов некоторого изображения: а) контур 1; б) контур 2

3. Правила идентификации угловой точки

Правила идентификации угловой точки подразделяются на следующие два типа: ложной угловой точки на основе аппроксимации описания бинарного контура объекта изображения и угловой точки на основе аппроксимации описания бинарного контура объекта изображения векторами \bar{a} и \bar{b} .

Как уже упоминалось выше, идентификация угловой точки выполняется на четвертом шаге работы рассматриваемого детектора угловых точек. Пусть известны отсчет-«претендент» на угловую точку и описания соответствующих «верхнего» и «нижнего» контуров в терминах 8-связного цепного кода Фримана. Тогда идентификация угловой точки может быть осуществлена в следующие три этапа: идентификация ложной угловой точки, т. е. отсчета-«претендента», не являющегося угловой точкой, посредством правил идентификации первого типа; аппроксимация описаний «верхнего» и «нижнего» контуров отсчета-«претендента» векторами \bar{a} и \bar{b} ; идентификация угловой точки посредством правил идентификации второго типа.

Идентификация угловой точки выполняется на основе аппроксимации имеющихся описаний «верхнего» и «нижнего» контуров для отсчета-«претендента», т. е. анализа количества того или иного направления в описании соответствующего контура. Отметим, что направления с кодом 0 или 2 в описании контуров при их анализе считаются промежуточными, а направления с кодом 1, 3, 5, 7 рассматриваются как положительные – угол образующие направления.

Рассмотрим правила идентификации угловой точки более подробно.

Правила идентификации первого типа. Отсчет-«претендент» не является угловой точкой при наличии осцилляций в описании «верхнего» или «нижнего» контура, т. е. направлений с кодом 0–2, 1–3, 5–7, 3–5, 1–7. Отсчет-«претендент» не является угловой точкой в том случае, если описание «верхнего» или «нижнего» контура аппроксимируется дугой. Отсчет-«претендент» не является угловой точкой в том случае, если в соответствующих угол обра-

зующих отсчетах происходит соприкосновение «верхнего» и «нижнего» контуров, т. е. у контуров отсутствует общий отсчет (точка O). Отсчет-«претендент» не является угловой точкой в том случае, если описания «верхнего» и «нижнего» контуров аппроксимируются направлениями с кодом 0–0, 2–2, 1–5, 3–7, 5–1, 7–3. Отсчет-«претендент» не является угловой точкой, если в начале описания «верхнего» или «нижнего» контура содержатся угол образующие отсчеты, принадлежащие другой угловой точке.

Правила идентификации второго типа. Отсчет-«претендент» является угловой точкой в том случае, если описания «верхнего» и «нижнего» контуров состоят только из одного направления с кодом 0 или 2, а также из направлений с кодом 0, 2 и одного из угол образующих направлений, причем присутствие угол образующего направления в описании соответствующего контура обязательно. Например, описание «верхнего» контура состоит из направлений с кодом 0, 1, 2, а описание «нижнего» контура – из направлений с кодом 0, 2, 7.

Аппроксимация описаний «верхнего» и «нижнего» контуров векторами \bar{a} или \bar{b} выполняется на основе следующих правил:

1. В случае если описание «верхнего» или «нижнего» контура состоит из направлений с кодом 0, 3, 5 или 0, 1, 7 и каждое из направлений с кодом 3, 5 или 1, 7 встречается в описании контура wk раз, то такое описание контура может быть заменено описанием, состоящим только из направления с кодом 0.

2. В случае если описание «верхнего» или «нижнего» контура состоит из направлений с кодом 2, 5, 7 или 1, 2, 3 и каждое из направлений с кодом 5, 7 или 1, 3 встречается в описании контура wk раз, то такое описание контура может быть заменено описанием, состоящим только из направления с кодом 2.

4. Результаты экспериментов

Представленный детектор угловых точек был протестирован на различных изображениях, как реальных, так и сгенерированных, также было произведено сравнение его работы с другими известными детекторами угловых точек.

Результаты тестирования рассматриваемого детектора угловых точек, детекторов Харриса и CSS показаны на рис. 5–7 и сведены в таблицу. Тестирование детекторов осуществлялось на базе персонального компьютера IBM PC Pentium IV 3.2 ГГц с 1024 Мб ОЗУ. Отметим, что в таблице представлено среднее время работы детекторов, полученное в результате 100 запусков, при этом исходные данные не изменялись. Тестирование рассматриваемого детектора осуществлялось со следующими значениями параметров: $n = 8$, $wk = 1$, $m_x = m_y = 3$. Детекторы Харриса [2] и CSS [3] тестировались с экспериментально подобранными значениями параметров, при которых детекторы показали наилучший результат в нахождении угловых точек, т. е. со значением параметра $\delta = 2$ и $\delta = 3$ соответственно. Фильтр Канни использовался со значениями порогов $T1 = 0$ и $T2 = 0,35$. Проведенные эксперименты показали также, что с уменьшением значения δ детекторы Харриса и CSS лучше определяют тупые углы, однако при этом количество найденных ложных угловых точек на острых углах существенно возрастает. С увеличением значения δ происходит уменьшение количества определяемых угловых точек.

Оценка работы рассматриваемых детекторов угловых точек может быть произведена на основе следующих коэффициентов:

$$k_1 = \frac{N}{R}; \quad (2)$$

$$k_2 = \frac{L}{R}, \quad (3)$$

где N – количество ненайденных угловых точек; L – количество определенных ложных угловых точек; R – общее число угловых точек объектов изображения (таблица).

Результаты тестирования рассматриваемых детекторов угловых точек

Рисунок	Найденные углы	Ненайденные углы	Ложные углы	Общее число углов	Время работы детектора, с	k_1	k_2
5, а	46	0	1	45	1,4641	0	0,02
5, б	42	0	1	41	1,2486	0	0,02
5, в	50	1	6	45	0,1478	0,02	0,13
5, г	35	6	0	41	0,0912	0,15	0
6, в	49	—	—	—	0,6842	—	—
6, г	76	—	—	—	0,1276	—	—
7, а	332	—	—	—	2,8991	—	—
7, б	243	—	—	—	2,9175	—	—
7, в	288	—	—	—	1,354	—	—
7, г	189	—	—	—	1,5793	—	—

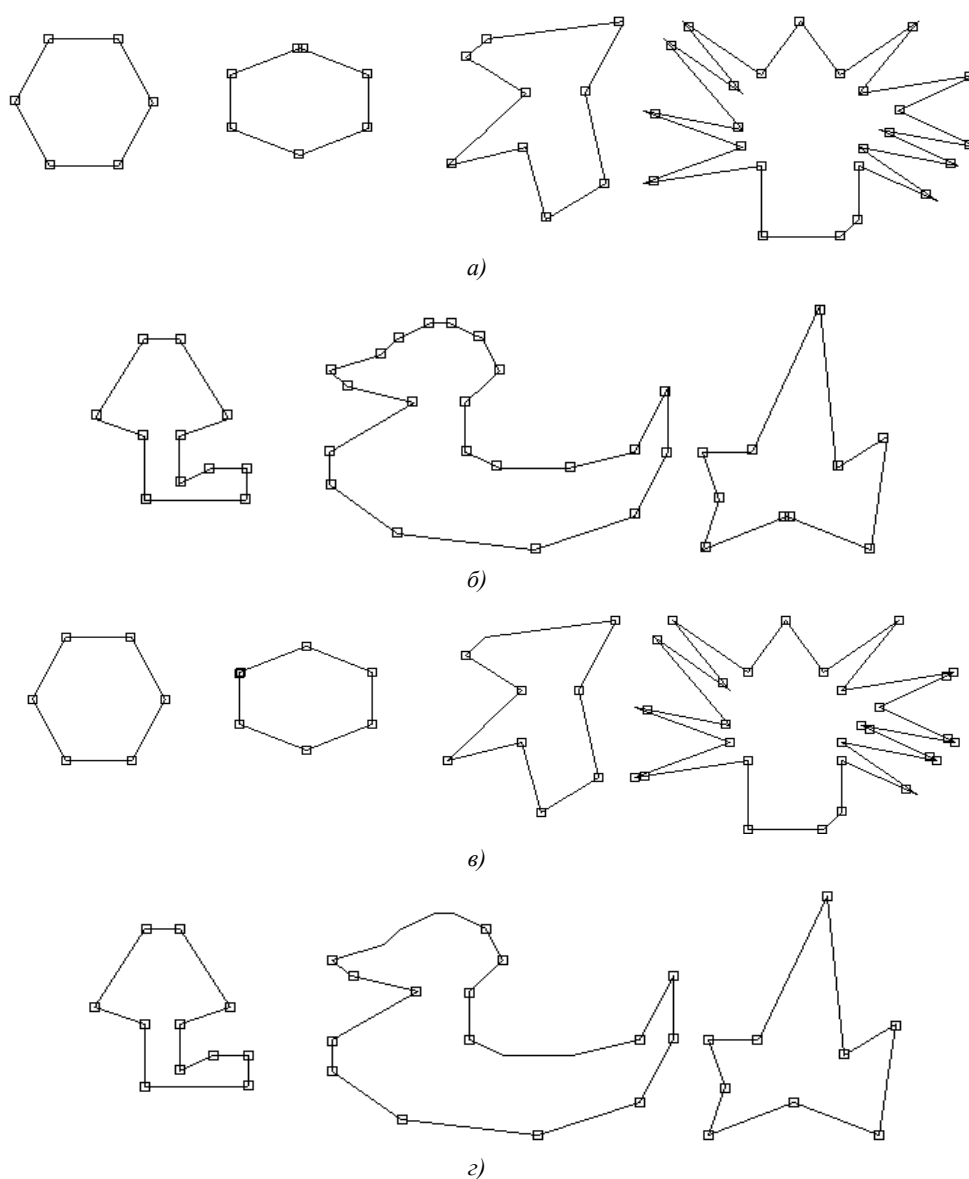


Рис. 5. Результаты тестирования детекторов угловых точек с использованием изображений I_1 и I_2 размером 717×209 и 594×224 пикселей: а) рассматриваемый детектор, $n = 8$; б) рассматриваемый детектор, $n = 8$; в), г) детектор CSS

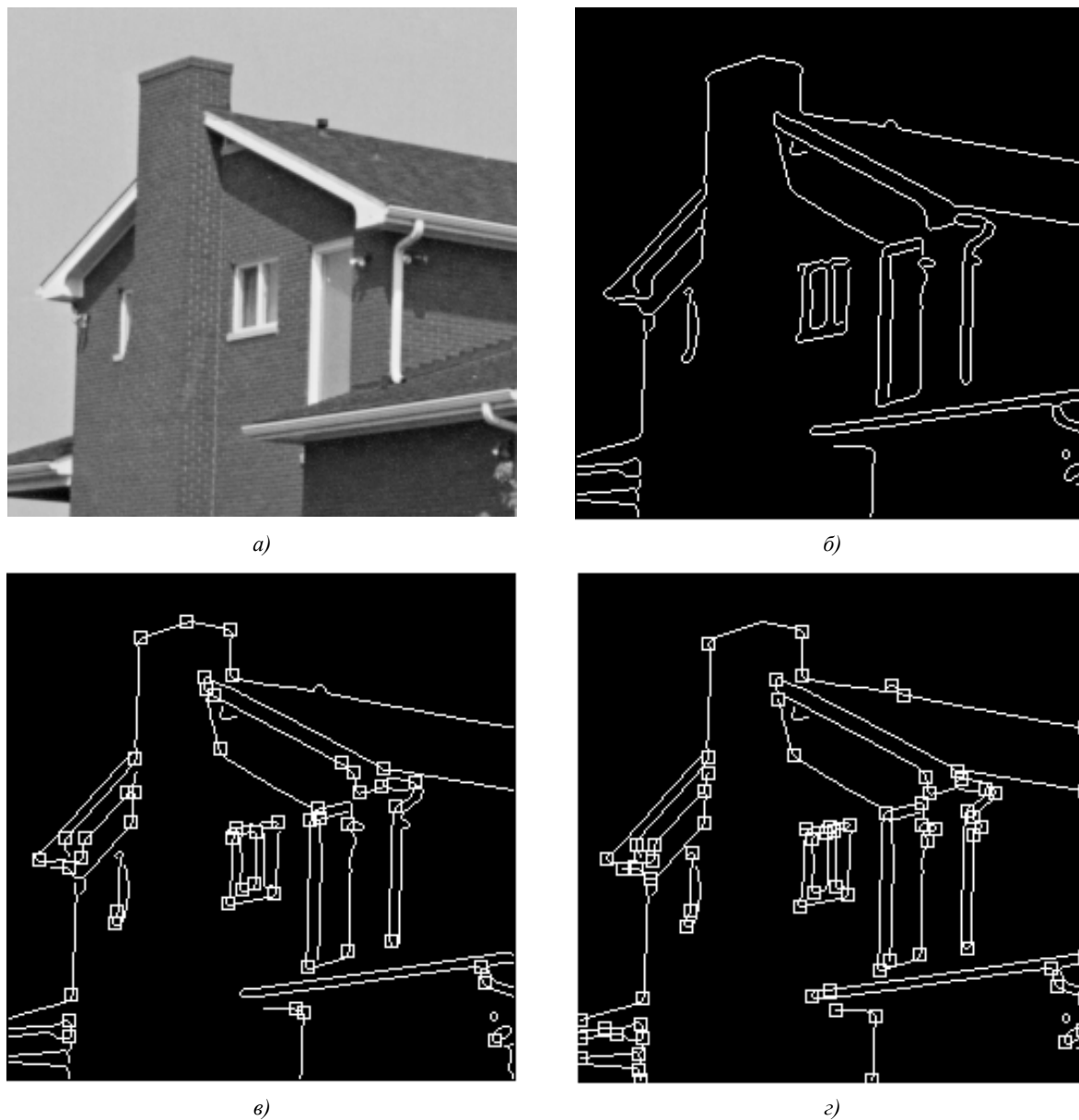


Рис. 6. Результаты тестирования детекторов угловых точек с использованием изображения «дом» размером 256×256 пикселей: а) исходное изображение; б) исходное изображение, представленное в бинарном виде; в) рассматриваемый детектор, $n = 8$; г) детектор CSS

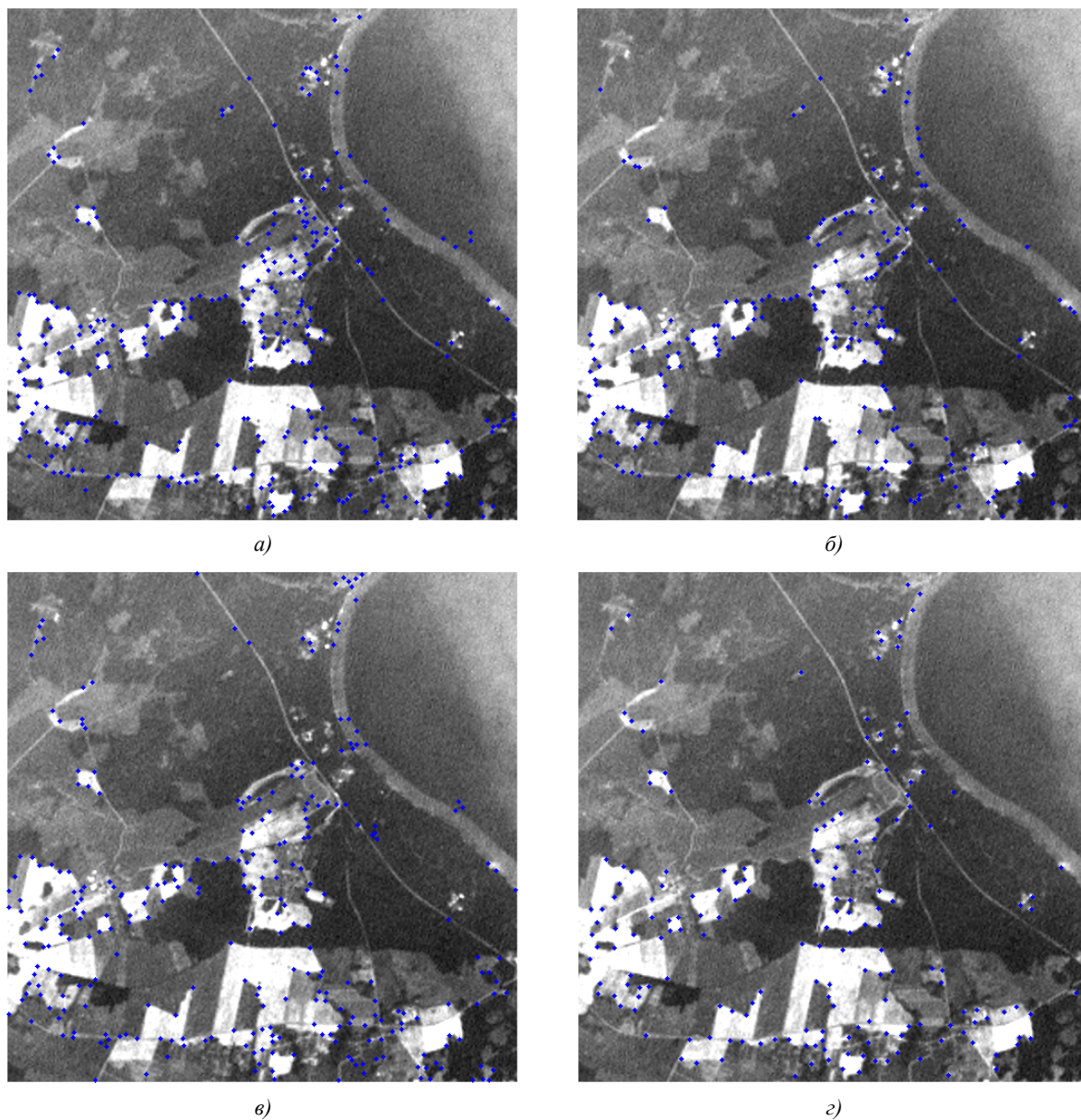


Рис. 7. Результаты тестирования детекторов угловых точек с использованием реального изображения I размером 501×501 пикселей: а) рассматриваемый детектор, $n = 7$; б) рассматриваемый детектор, $n = 9$; в) детектор CSS; г) детектор Харриса

Стоит также отметить, что представленный детектор угловых точек успешно используется в задачах взаимного совмещения изображений, которые, в свою очередь, находят применение в задачах дистанционного зондирования (землепользования, прогнозирования урожая, наблюдения за лесными ресурсами и др.).

Заключение

В статье рассмотрен детектор угловых точек на основе аппроксимации контуров объектов изображения. В основе детектора лежат следующие четыре шага: получение бинарных краев объектов исходного изображения, нахождение отсчета-«претендента» на угловую точку, построение описания бинарного контура объекта изображения для отсчета-«претендента» в терминах цепного кода Фримана, идентификация угловой точки.

Эксперименты показали, что с увеличением количества бинарных контуров объектов на рассматриваемых изображениях предложенный детектор угловых точек становится сопоставимым с конкурирующими с ним детекторами угловых точек как по качеству нахождения угловых точек, так и по быстродействию.

Список литературы

1. Moravec, H.P. Towards automatic visual obstacle avoidance / H.P. Moravec // Proc. of the 5th Intern. Joint Conf. of Artificial Intelligence. – Cambridge, 1977. – P. 587–598.
2. Harris, C. A combined corner and edge detector / C. Harris, M. Stephens // Proc. of the Fourth Alvey Vision Conf. – UK, 1988. – P. 147–152.
3. Mokhtarian, F. Robust image corner detection through curvature scale space / F. Mokhtarian, R. Suomela // IEEE Trans. On Pattern Analysis and Machine Intelligence. – 1998. – Vol. 20, № 12. – P. 1376–1381.
4. Mokhtarian, F. Enhancing the curvature scale space corner detector / F. Mokhtarian, F. Mohanna // Proc. of the Scandinavian Conf. on Image Analysis. – Sweden, 2001. – P. 145–152.
5. He, X.C. Corner detector based on global and local curvature properties / X.C. He, N.H.C. Yung // Optical Engineering. – 2008. – Vol. 47, № 5. – P. 057008-1–057008-12.
6. Методы компьютерной обработки изображений / под ред. В.А. Сойфера. – М. : Физматлит, 2003. – 784 с.
7. Выгодский, М.Я. Справочник по математике / М.Я. Выгодский. – М. : Астрель, 2010. – 1055 с.
8. Canny, J. Computational Approach to Edge Detection / J. Canny // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 1986. – Vol. 8, № 6. – P. 679–698.
9. Введение в контурный анализ: приложения к обработке изображений и сигналов / Я.А. Фурман [и др.]. – М. : Физматлит, 2003. – 588 с.
10. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1072 с.

Поступила 10.11.09

*Белорусский государственный университет
информатики и радиоэлектроники,
Минск, ул. П. Бровки, 6
e-mail: anton.kazlouski@yahoo.com*

A.N. Kazlouski

**A CORNER DETECTOR BASED ON APPROXIMATION
OF CONTOURS OF IMAGE OBJECTS**

A corner detector based on approximation of contours of image objects is presented. The corner detector implements the following chain of operations: significant point detection, description of binary contours of image object in terms of Freeman code, applying salient point identification rules based on approximation of constructed contour description. The proposed corner detector was compared to the conventional detectors. The results have proved the efficiency of suggested filter.