

УДК 621.391

СЕКТОРНАЯ ЛОКАЛИЗАЦИЯ, ПАРАМЕТРИЗАЦИЯ И ИДЕНТИФИКАЦИЯ РЕПЕРОВ НА ОСНОВЕ УГЛОВЫХ КОЭФФИЦИЕНТОВ ДЛЯ СОВМЕЩЕНИЯ ПЕРЕКРЫВАЮЩИХСЯ ИЗОБРАЖЕНИЙ

О.ДЖ. АЛЬ-ФУРАЙДЖИ, В.К. КОНОПЕЛЬКО, В.Ю. ЦВЕТКОВ

Белорусский государственный университет информатики и радиоэлектроники
П.Бровки, 6, Минск, 220013, Беларусь

Поступила в редакцию 9 июля 2012

Предложены алгоритмы секторной локализации и параметризации реперов в вейвлет-области для совмещения перекрывающихся изображений, основанные на использовании угловых коэффициентов для описания контурной структуры в окрестности репера. Показано, что данные алгоритмы по сравнению с методами *SIFT* и *SURF* обеспечивают снижение вычислительной сложности локализации и параметризации реперов и повышение стабильности реперных идентификаторов в условиях параллакса.

Ключевые слова: локализация и параметризация реперов, совмещение изображений.

Введение

В основе совмещения перекрывающихся изображений лежит поиск соответствий. Он основан на локализации и параметризации реперов на изображении. Известные методы локализации и параметризации реперов *SIFT* (*Scale Invariant Feature Transform*) [1] и *SURF* (*Speeded-Up Robust Features*) [2] используют иерархию аппроксимированных представлений изображения для локализации реперов в различных масштабах с последующей параметризацией, обеспечивающей их идентификацию инвариантно относительно поворота, масштабирования, изменения яркости изображений. Использование большого числа разномасштабных представлений изображения для локализации реперов приводит к высокой вычислительной сложности, что является основным недостатком данных методов. Кроме того, идентификаторы в методах *SIFT* и *SURF* не учитывают параллакс, возникающий при смещении видеокамеры.

Целью работы является разработка вычислительно простых алгоритмов локализации и параметризации реперов для совмещения перекрывающихся изображений в условиях параллакса.

Выделение реперов

Локализация реперов начинается с их выделения на изображении. Для снижения вычислительной сложности предлагается алгоритм *HRT* (*Haar Roberts Transform*) формирования реперных образов на основе контурной бинаризации с использованием преобразований Хаара [3] и Робертса [4], включающий три шага: формирование кратномасштабного представления изображения с помощью вейвлет-преобразования Хаара с одним уровнем аппроксимации на каждом уровне кратномасштабного представления (6 целочисленных операций сложения на пиксель); контурная фильтрация аппроксимированных образов с помощью фильтра Робертса (3 целочисленных операции сложения на пиксель); формирование множества $\{R(l_S, l_R)\}_{(l_S=0, \overline{L_S-1}, l_R=1)}$ бинарных реперных образов на основе пороговой обработки результатов контурной фильтрации (1 целочисленная операция сложения на пиксель). Алгоритм *HRT* ис-

пользует только целочисленные операции сложения, число $C_{HRT}^{PP}(YX)$ которых вычисляется с помощью выражения

$$C_{HRT}^{PP}(YX) = 4YX + 10 \sum_{l_s=0}^{L_s-1} \frac{YX}{2^{2(l_s+1)}}, \quad (1)$$

где L_s – число уровней кратномасштабного представления изображения; YX – размер изображения.

На рис. 1 а приведены зависимости $C_{SIFT}^{PP}(YX)$, $C_{SURF}^{PP}(YX)$ и $C_{HRT}^{PP}(YX)$ от размера YX изображения при $L_s = 4$. Из рис. 1 а следует, что алгоритм *HRT* имеет в 5 раз меньшую вычислительную сложность по сравнению с методом *SURF*.

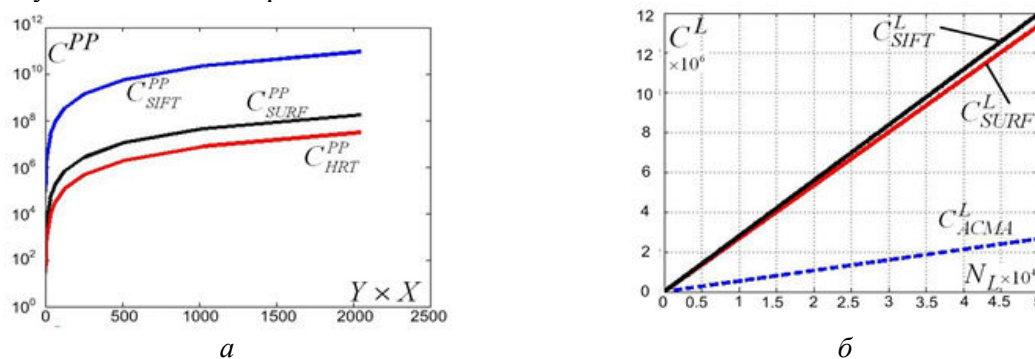


Рис. 1. Вычислительная сложность:
а – выделения реперов; б – уточнения реперов

Уточнение и секторная локализация реперов

Реперные образы содержат множество точек, которые не могут быть использованы в качестве реперов из-за отсутствия в их окрестности признаков, обеспечивающих однозначную идентификацию. Для локализации подходящих точек необходим этап уточнения реперов.

Для снижения вычислительной сложности этапа локализации реперов и повышения устойчивости его результатов к изменению яркости и масштаба изображения предлагается алгоритм *ACMA* (*Angle Contour Mask Algorithm*) уточнения реперов, основанный на использовании библиотеки из 24-х угловых контурных масок, представляющих все возможные повороты на угол $45d^\circ$ трех базовых угловых контурных масок размером 4×4 пикселей со значениями угла $\alpha = \{45^\circ, 90^\circ, 135^\circ\}$ (рис. 2), где $d = 0,7$ – ориентация. Библиотека применяется для поиска угловых реперов на множестве $\{R(l_s, l_r)\}_{(l_s=0, L_s-1, l_r=1)}$ реперных образов. Совместное использование алгоритмов *HRT* и *ACMA* обеспечивает секторную локализацию реперов, позволяющую определить местоположение угловых контурных реперов, окрестность которых разделена контуром на два сектора, угловые размеры и ориентация которых определяются с помощью одной из 24 бинарных угловых контурных масок. Секторная локализация реперов инвариантна к повороту, изменению яркости и масштаба изображения, т.к. угловые параметры секторов слабо зависят от этих преобразований.

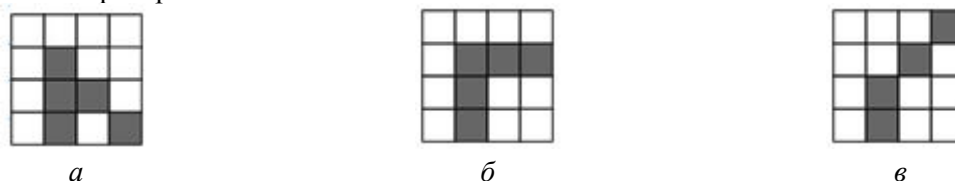


Рис. 2. Базовые бинарные угловые контурные маски:
а – маска 45° ; б – маска 90° ; в – маска 135°

Алгоритм уточнения реперов состоит из следующих двух шагов.

1. Окрестность каждого репера из множества $\{R(l_S, l_R)\}_{(l_S=0, \overline{L_S-1}, l_R=1)}$ проверяется на соответствие одной из 24 угловых контурных масок. Это требует $24 \times 4 \times 4 = 384$ бинарные операции сложения. В [6] показано, что за счет минимизации числа узлов поискового графа количество бинарных операций может быть сокращено до 52. В результате формируется множество $\{C(l_S, l_R)\}_{(l_S=0, \overline{L_S-1}, l_R=1)}$ угловых реперных образов. Для каждого репера $c(l_S, l_R, n_{l_S})$ из углового реперного образа $C(l_S, l_R)$ фиксируется набор параметров $\{y(n_{l_S}), x(n_{l_S}), \alpha(n_{l_S}), d(n_{l_S}), l_S(n_{l_S})\}_{(n_{l_S}=0, \overline{N_C(l_S)})}$, где $N_C(l_S)$ – число реперов на уровне l_S .

2. Отыскиваются пары соответствующих угловых контурных реперов $c(l_S, l_R, i)$ и $c(l_S - 1, l_R, j)$, удовлетворяющих условию

$$\forall l_S (l_S = \overline{L_S, 1}) \forall i (i = \overline{0, N_C(l_S)}) \exists j (j \in [0, N_C(l_S - 1)]) \left(\begin{aligned} & \left((2y(i) \approx y(j)) \wedge (2x(i) \approx x(j)) \wedge \right. \\ & \left. \wedge (\alpha(i) \approx \alpha(j)) \wedge (d(i) \approx d(j)) \right) \end{aligned} \right) \rightarrow (2) \\ \rightarrow (c(l_S, l_R, i) \Leftrightarrow c(l_S - 1, l_R, j)),$$

где $\Leftrightarrow$ – символ соответствия угловых контурных реперов на соседних вейвлет-уровнях.

В результате данного шага выбираются угловые контурные реперы, локализация которых инвариантна к изменению масштаба изображения в 2 раза. Если допустимое отклонение для каждого из четырех параметров $\{y(j), x(j), \alpha(j), d(j)\}$ составляет 1 дискретное значение, то второй шаг требует 15 целочисленных операций сложения на каждый угловой репер.

Таким образом, для уточнения N_L реперов алгоритм АСМА требует $C_{АСМА}^L(N_L)$ целочисленных операций сложения, определяемых с помощью выражения

$$C_{АСМА}^L(N_L) = N_L (52/f_{IBA} + 15K_C), \quad (3)$$

где f_{IBA} – коэффициент, учитывающий сложность целочисленной операции сложения по отношению к бинарной операции сложения; K_C – коэффициент, определяющий долю угловых реперов на верхних $l_S = \overline{1, L_S - 1}$ уровнях по отношению к общему числу выделенных реперов.

На рис. 1 б приведены зависимости вычислительной сложности $C_{SIFT}^L(N_L)$, $C_{SURF}^L(N_L)$ и $C_{АСМА}^L(N_L)$ от числа N_L реперов при условии $K_C = 0,07$ (значение коэффициента определено экспериментально в результате усреднения по коллекции тестовых изображений с различной статистикой распределения яркости). Значение коэффициента $f_{IBA} = 1$ определено для процессора Intel Pentium [6]. Из рис. 1 б следует, что алгоритм АСМА имеет в 5 раз меньшую вычислительную сложность по сравнению с методами SIFT и SURF.

Секторная параметризация реперов

Для снижения вычислительной сложности этапа параметризации реперов и повышения устойчивости его результатов к параллаксу предлагается алгоритм секторной параметризации реперов (SPRIT – Scale Parallax Rotate Invariant Transform) на основе угловых коэффициентов, вычисляемых для контурных точек в окрестности репера.

Алгоритм секторной параметризации состоит из следующих 3-х шагов.

1 Для каждого репера с координатами (y_{MP}, x_{MP}) формируется вектор $D_{MP}(y_{MP}, x_{MP}, R_{MP}) = \|d_{MP}(y_{MP}, x_{MP}, R_{MP}, i_{CP})\|_{(i_{CP}=\overline{1, N_{CP}(y_{MP}, x_{MP}, R_{MP})})}$, в котором значению i_{CP} -го элемента соответствует расстояние от этого репера до i_{CP} -ой контурной точки, находящейся в его окрестности радиусом R_{MP} , где $N_{CP}(y_{MP}, x_{MP}, R_{MP})$ – число контурных точек в окрестности ре-

пера радиусом R_{MP} . При условии, что число N_L контурных пикселей и общее число YX пикселей связано отношением $K_p = N_L / (YX)$ и локальная плотность контурных пикселей для всего изображения одинакова, на данном шаге выполняются $3K_p\pi R_{MP}^2$ целочисленных операций сложения, $2K_p\pi R_{MP}^2$ целочисленных операций умножения и $K_p\pi R_{MP}^2$ вещественных операций извлечения корня квадратного на репер.

2) В окрестности каждого репера выделяются ядро радиусом $R_{MP}/2$ пикселей; $R_{MP}/2$ орбит шириной в 1 пиксель на расстояниях $\overline{R_{MP}/2 + 1, R_{MP}}$ пикселей от репера; внутренний S_{In} и внешний S_{Out} секторы, размеры и ориентация которых определяются по одному из 24 угловых шаблонов, которому соответствует репер. Данные секторы делятся далее еще на три части каждый (на секторы $S_{In}^{(M)}, S_{In}^{(B)}, S_{In}^{(E)}$ – для сектора S_{In} ; на секторы $S_{Out}^{(M)}, S_{Out}^{(B)}, S_{Out}^{(E)}$ – для сектора S_{Out}) в результате выделения по краям секторов размером $\pi/8$ (рис. 3).

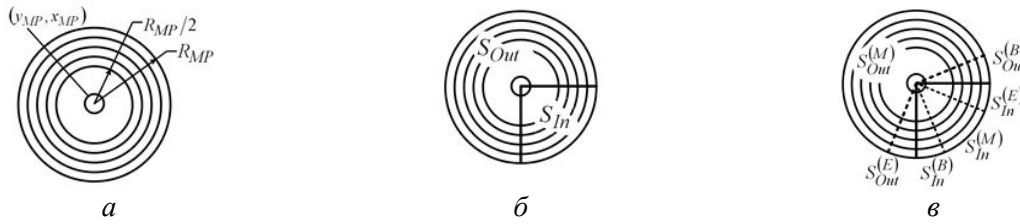


Рис. 3. Выделение секторов в окрестности репера:

a – формирование ядра и орбит; *б* – формирование внутреннего и внешнего секторов;
в – формирование дополнительных секторов

3. Рассматриваются три пары секторов: секторы S_{In} и S_{Out} (пара 1); секторы $(S_{Out}^{(E)} \cup S_{In}^{(B)} \cup S_{In}^{(M)})$ и $(S_{In}^{(E)} \cup S_{Out}^{(B)} \cup S_{Out}^{(M)})$ (пара 2); секторы $(S_{In}^{(M)} \cup S_{In}^{(E)} \cup S_{Out}^{(B)})$ и $(S_{Out}^{(M)} \cup S_{Out}^{(E)} \cup S_{In}^{(B)})$ (пара 3), где символом $\cup$ обозначена операция получения составных секторов. Для каждого сектора строится гистограмма $H_{CT}^{(ST)} = \|h_{CT}^{(ST)}(i)\|_{i=0, N_{CT}^{(ST)}}$, значения элементов которой представляют взвешенную статистику угловых коэффициентов для прямых, проходящих через репер и пары контурных пикселей, которые удалены друг от друга на угловое расстояние S_W и от репера на расстояние $d_{MP}(y_{MP}, x_{MP}, R_{MP}, i_{CP})$, удовлетворяющее условию

$$d - 1 < d_{MP}(y_{MP}, x_{MP}, R_{MP}, i_{CP}) < d + 1 \quad (4)$$

при $d = \overline{2, R_{MP} - 1}$, где CT и ST – индексы, указывающие на принадлежность гистограммы к конкретному сектору $S_{CT}^{(ST)}$ и принимающие значения $\{In, Out\}$ и $\{B, M, E\}$ соответственно.

Число элементов $N_{CT}^{(ST)}$ в гистограмме $H_{CT}^{(ST)}$ вычисляется с помощью выражения

$$N_{CT}^{(ST)} = \begin{cases} S_{CT}^{(ST)} / p & \text{при } S_{CT}^{(ST)} < S_W, \\ S_W / p & \text{при } S_{CT}^{(ST)} \geq S_W, \end{cases} \quad (5)$$

где $p = \begin{cases} p_1 & \text{при } CT = In, \\ p_2 & \text{при } CT = Out \end{cases}$ – точность гистограммы, определяемая числом сегментов, которые она содержит, и сектором, для которого она строится ($p_1 > p_2$).

При условии, что контурные пиксели равномерно распределены по окрестности репера, вычисление статистики ограничено сектором S_W и тремя орбитами согласно условию (4), количество угловых расстояний, которые должны быть вычислены между контурными точками, составляет $(R_{MP} - 1)(3S_W K_p \pi R_{MP}^2 / (360 R_{MP}))^2 = (R_{MP} - 1)(S_W K_p R_{MP})^2 / 1444$. Тогда, для вычис-

ления угловых расстояний необходимо $(R_{MP}-1)(S_W K_P R_{MP})^2 / 481$ целочисленных операций сложения, $(R_{MP}-1)(S_W K_P R_{MP})^2 / 722$ целочисленных операций умножения, $(R_{MP}-1)(S_W K_P R_{MP})^2 / 1444$ вещественных операций извлечения корня квадратного и столько же вещественных операций деления на репер. Для взвешивания статистики используется K_G дискретных значений одной из $R_{MP}-2$ функций Гаусса с единичной площадью и различной шириной, постепенно уменьшающейся в направлении от репера к границам рассматриваемой окрестности. Формирование взвешенной гистограммы требует выполнения $K_G (R_{MP}-1)(S_W K_P R_{MP})^2 / 1444$ вещественных операций сложения и столько же вещественных операций умножения на репер.

В результате данного алгоритма для каждого репера формируются 6 идентификаторов, каждый из которых может быть использован для поиска соответствия перекрывающихся изображений независимо, что обеспечивает инвариантность идентификации к параллаксу.

Для алгоритма *SPRIT* количество $C_{SPRIT}^P(N_F)$ вещественных и целочисленных операций, выраженное в целочисленных операциях сложения, вычисляется с помощью выражения

$$C_{SPRIT}^P(N_F) = N_F \left(K_P \pi R_{MP}^2 (5 + f_{FPIA} f_{FPSA}) + 1/481 + f_{FPIA} f_{FPSA} / 1444 + \right. \\ \left. + (R_{MP}-1)(S_W K_P R_{MP})^2 ((1 + f_{FPIA} f_{FPDM} / 2) / 722) \right), \quad (6)$$

где N_F – число реперов; f_{FPIA} – коэффициент, учитывающий вычислительную сложность вещественной операции сложения по отношению к целочисленной операции сложения; f_{FPIA} – коэффициент, учитывающий сложность вещественной операции умножения по отношению к целочисленной операции умножения; f_{FPSA} – коэффициент, учитывающий сложность вещественной операции извлечения корня квадратного по отношению к вещественной операции сложения; f_{FPDM} – коэффициент, учитывающий сложность вещественной операции деления по отношению к вещественной операции умножения.

На рис. 4 а приведены зависимости вычислительной сложности $C_{SIFT}^P(N_F)$, $C_{SURF}^P(N_F)$ и $C_{SPRIT}^P(N_F)$ от числа N_F реперов при условии $K_P = 0,07$ (определено экспериментально в результате усреднения по коллекции тестовых изображений с различной статистикой распределения яркости). Значения коэффициентов $f_{FPIA} = 3$, $f_{FPIA} = 0,6$, $f_{FPSA} = 23,3$, $f_{FPDM} = 7$ определены для процессора *Intel Pentium* [5]. Из рис. 4 а следует, что предложенный алгоритм *SPRIT* секторной параметризации реперов имеет в 18 раз и 43 раза меньшую вычислительную сложность по сравнению с методами *SIFT* и *SURF* соответственно.

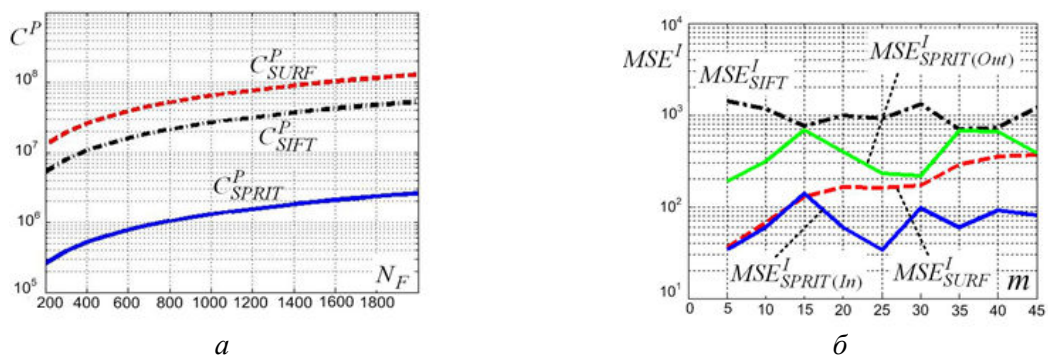


Рис. 4. Оценка эффективности секторной параметризации реперов:

а – зависимости вычислительной сложности C^P этапа параметризации реперов от числа реперов;

б – зависимости ошибки MSE^I идентификации реперов от смещения видеокамеры

На рис. 4 б представлены зависимости среднеквадратической ошибки MSE^I идентификации реперов от величины m горизонтального смещения (в см) видеокамеры ($MSE_{SPRIT(In)}^I$ – для In-сектора в алгоритме *SPRIT*; для метода *SIFT*; $MSE_{SPRIT(Out)}^I$ – для Out-сектора в алгоритме *SPRIT*; MSE_{SIFT}^I – для метода *SIFT*; MSE_{SURF}^I – для метода *SURF*). Данные зависимости характеризуют устойчивость результатов параметризации реперов к параллаксу и получаются в результате сопоставления идентификаторов реперов на исходном изображении и изображениях, полученных при смещении видеокамеры. Из рис. 4 б следует, что алгоритм *SPRIT* секторной параметризации позволяет повысить стабильность идентификаторов в условиях параллакса до 10 раз и 4,4 раза по сравнению с методами *SIFT* и *SURF* соответственно.

Оценка эффективности секторной идентификации реперов

Для оценки вычислительной сложности C^{LP} локализации и параметризации реперов на основе алгоритмов *HAS* (*HRT*, *ACMA*, *SPRIT*) (C_{HAS}^{LP}) и методов *SIFT* (C_{SIFT}^{LP}), *SURF* (C_{SURF}^{LP}) предлагается использовать следующее выражение

$$C_A^{LP} = C_A^{PP}(YX) + C_A^L(k_A^{(L)}YX) + C_A^P(k_A^{(P)}YX), \quad (7)$$

где A – индекс, указывающий на алгоритм (метод) и принимающий значение из множества $\{HAS, SIFT, SURF\}$; $k_A^{(L)}$ и $k_A^{(P)}$ – коэффициенты, определяющие отношение числа контурных точек и реперов соответственно, выделяемых с помощью алгоритма A , к общему количеству YX пикселей изображения.

Оценка вычислительной сложности идентификации реперов для алгоритмов *HAS* (C_{HAS}^I) и методов *SIFT* (C_{SIFT}^I), *SURF* (C_{SURF}^I) производится с помощью выражений

$$C_A^I = 9R_A(k_A^{(P)}YX)^2, \quad (8)$$

где R_A – число элементов в идентификаторах для алгоритма A .

Для алгоритмов *HAS* и методов *SIFT*, *SURF* в табл. 1 приведены значения $k_A^{(L)}$, $k_A^{(P)}$, C_A^{LP} и C_A^I для тестового ландшафтного изображения. Из табл. 1 следует, что алгоритмы *HAS* обеспечивают по сравнению с методом *SURF* уменьшение до 5,5 раз вычислительной сложности локализации и параметризации реперов и до 4 раз вычислительной сложности идентификации реперов. По отношению к методу *SURF* суммарная вычислительная сложность $C^{LP} + C^I$ секторной локализации, параметризации и идентификации реперов уменьшается в 4 раза.

Таблица 1. Вычислительная сложность локализации, параметризации и идентификации реперов

Алгоритмы (методы)	$k_A^{(L)}$	$k_A^{(P)}$	$C_A^{LP} \times 10^8$	$C_A^I \times 10^8$
<i>HAS</i>	0,1361	0,0018	0,023	0,991
<i>SIFT</i>	0,0095	0,0021	64,933	1,598
<i>SURF</i>	0,2116	0,0044	0,128	3,508

Для алгоритмов *HAS* и методов *SIFT*, *SURF* в табл. 2 приведены зависимости числа $N_I(m)$ идентифицированных и $\tilde{N}_I(m)$ неидентифицированных реперов от сдвига m видеокамеры по горизонтали, полученные с помощью метода *RANSAC* [7]. Из табл. 2 следует, что с увеличением смещения видеокамеры число ($N_I(m) + \tilde{N}_I(m)$) реперов на этапе идентификации для методов *SIFT* и *SURF* резко сокращается, а для предложенных алгоритмов *HAS* секторной локализации и параметризации остается примерно постоянным. Алгоритмы *HAS* обеспечивают минимальное значение R_{HAS} отношения дисперсий величин $N_I(m)$ и $\tilde{N}_I(m)$ ($R_{HAS} = 0,25$) по сравнению с *SIFT* ($R_{SIFT} = 0,63$) и *SURF* ($R_{SURF} = 1,98$), характеризующее устойчивость идентификации реперов к параллаксу. Из табл. 2 следует, что при сдвиге видеокамеры до 45 см сек-

торная идентификация реперов на базе алгоритмов *HAS* в 2,5 раза и в 8 раз более устойчива к параллаксу по сравнению с идентификацией на базе методов *SIFT* и *SURF* соответственно.

Таблица 2. Значения $\frac{N_I(m)}{\tilde{N}_I(m)}$ числа идентифицированных и неидентифицированных реперов

Алгоритмы, методы	Значения сдвига m видеокамеры, см								
	5	10	15	20	25	30	35	40	45
<i>HAS</i> ($R_{HAS} = 0,25$)	$\frac{82}{307}$	$\frac{33}{263}$	$\frac{40}{234}$	$\frac{22}{277}$	$\frac{24}{491}$	$\frac{31}{501}$	$\frac{32}{385}$	$\frac{41}{466}$	$\frac{30}{423}$
<i>SIFT</i> ($R_{SIFT} = 0,63$)	$\frac{52}{169}$	$\frac{43}{210}$	$\frac{26}{191}$	$\frac{33}{162}$	$\frac{19}{198}$	$\frac{18}{192}$	$\frac{12}{217}$	$\frac{8}{181}$	$\frac{10}{147}$
<i>SURF</i> ($R_{SURF} = 1,98$)	$\frac{271}{171}$	$\frac{225}{148}$	$\frac{162}{132}$	$\frac{139}{120}$	$\frac{77}{108}$	$\frac{60}{96}$	$\frac{73}{115}$	$\frac{56}{91}$	$\frac{53}{61}$

Заключение

Предложены алгоритмы секторной локализации и параметризации реперов полутонных изображений в вейвлет-области, основанные на использовании угловых коэффициентов для описания локальной контурной структуры в окрестности репера на исходном изображении и его аппроксимированных вейвлет-образах. Показано, что предложенные алгоритмы обеспечивают уменьшение вычислительной сложности локализации, параметризации и идентификации реперов в 4 раза по сравнению с методом *SURF* и повышение стабильности идентификации при параллаксе в 2,5 раза по сравнению с методом *SIFT*.

SECTOR LOCALIZATION, PARAMETERIZATION AND IDENTIFICATION OF REFERENCE POINTS BASED ON CORNER COEFFICIENTS FOR ALIGNING THE OVERLAPPING IMAGES

O.J. AL-FURAIJI, V.K. KANAPELKA, V.YU. TSVIATKOU

Abstract

Algorithms of sector localization and parameterization of reference points in the wavelet domain for aligning the overlapping images based on the utility of corner coefficients to describe the contour structure in the vicinity of the reference point are proposed. It is shown that these algorithms compared to the methods of SIFT and SURF provide a reduction in the computational complexity of localization and parameterization of reference points and increase the stability of their descriptors at parallax.

Список литературы

1. Lowe D.G. // International Journal of Computer Vision. 2004, Vol. 60, № 2. P. 91–110.
2. H. Bay, et. al // Computer Vision and Image Understanding. 2008, Vol. 110, № 3. P. 346–359.
3. Малла С. Вэйвлеты в обработке сигналов. М., 2005.
4. Гонсалес Р., Вудс Р. Цифровая обработка изображений. М., 2005.
5. Fog A. Optimization manuals, 4th, Instruction tables: Lists of instruction latencies, throughputs and micro-operation. Copenhagen University College of Engineering. 2012.
6. Аль-Фурайджи О.Дж., Спичекова Н.В. // Сб. междунар. научн.-техн. семинара. Телекоммуникации: сети и технологии, алгебраическое кодирование и безопасность данных. 2010. С. 51–56.
7. Fischler M.A., Bolles R.C. // Communications of the ACM. 1981., Vol. 24, № 6. – P. 381–395.