

ИДЕНТИФИКАЦИЯ И РАСЧЕТ ТРАЕКТОРИЙ ДИНАМИЧЕСКИХ ОБЪЕКТОВ ПО СТЕРЕОИЗОБРАЖЕНИЯМ

© 2020 г. В. А. Бобков^{а,*}, А. П. Кудряшов^{а,**}

^а *Институт автоматизи и процессов управления Дальневосточного отделения РАН
690041 Владивосток, ул. Радио, 5, Россия*

**E-mail: bobkov@dvo.ru*

***E-mail: kudryashovA@dvo.ru*

Поступила в редакцию 03.06.2019 г.

После доработки 17.06.2019 г.

Принята к публикации 23.06.2019 г.

Предложен модифицированный подход к решению задачи восстановления траекторий динамических объектов в сцене по стереоизображениям, основанный на применении расширенного точечного представления объектов, метода визуальной одометрии и оригинальной алгоритмической базы. Описаны алгоритмы идентификации объектов и вычислительная схема поэтапной обработки данных. Приведены результаты вычислительных экспериментов на модельных сценах.

DOI: 10.31857/S0132347420010021

1. ВВЕДЕНИЕ

В настоящей статье представлены результаты дальнейшего развития подхода к решению проблемы 3D SLAM в динамической сцене, опубликованной авторами в [1]. Некоторое представление о состоянии дел в этом направлении можно получить из работ [2–12], краткий обзор которых сделан в [1]. Была отмечена недостаточная степень универсальности существующих решений и не всегда приемлемая эффективность программно-алгоритмических средств для практических приложений.

В [1] был изложен подход, основанный на применении визуальной одометрии, использовании точечной модели объектов и реализации оригинальной алгоритмической базы для идентификации и обработки статических и динамических объектов (ДО) сцены без априорного знания о “статике”. Под ДО подразумеваются движущиеся объекты с непредсказуемыми моментами времени появления и исчезновения в сцене. Термин “статика” объединяет все неподвижные объекты в сцене. Однако полученные оценки эффективности первой реализации подхода указали на необходимость совершенствования разработанной алгоритмической базы. Основным недостатком заключался в генерации недостаточного числа точек в формируемых 3D моделях динамических объектов, что сказывалось на точности локализации объектов и качестве реконструируемых поверхностей объектов.

Поэтому в настоящей статье предлагается усовершенствованная вычислительная схема и алгоритмическая база с реализацией новых решений, направленных на повышение эффективности данного подхода к решению задачи SLAM для динамических сцен. К основным элементам новизны следует отнести использование плотного множества 3D точек, расширенного за счет измерений 3D сканера (или его виртуального аналога на базе стереоизображений) и возможность различать динамические объекты с неизменяемой и изменяемой формой.

2. ОБЩИЙ ПОДХОД

Решается задача идентификации и восстановления движения динамических объектов в априори неизвестной обстановке по изображениям, фиксируемым стереокамерой, установленной на автономном роботе. Предполагается, что в дополнение к оптической информации поступают и данные от 3D сканера (или от его виртуального аналога – программной реализации на базе стереокамеры [13]).

Предлагаемый подход к решению указанной задачи основывается на:

- применении точечного представления объектов сцены;
- применении визуального метода навигации для вычисления относительных перемещений камеры и объектов, и расчета их траекторий движения;

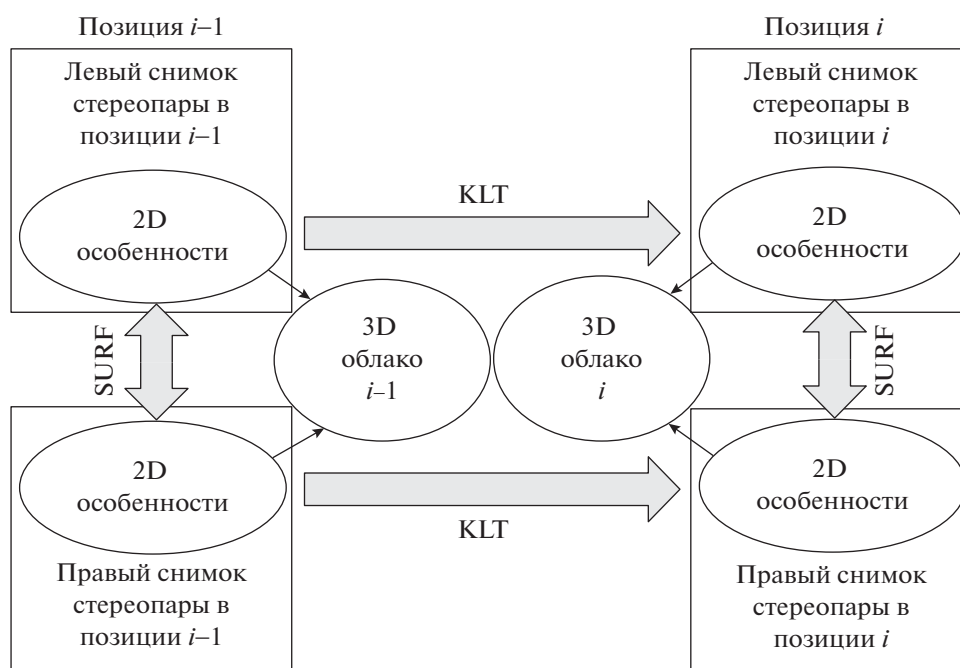


Рис. 1. Выделение единого сопоставленного множества особенностей на изображениях двух стереопар. Формирование 2-х 3D облаков.

– разработке алгоритмической базы, направленной на решение ключевых задач рассматриваемой проблемы, а именно:

– выделение ДО и генерация их 3D точечных моделей,

– выделение статической части сцены и вычисление траектории движения камеры,

– идентификация ДО, сформированных в разные моменты времени (в разных позициях траектории движения камеры),

– вычисление траекторий собственного движения ДО на основе декомпозиции комплексного движения объектов, фиксируемого камерой (движение камеры плюс собственное движение объекта).

Предлагаемое в настоящей работе развитие ранее описанного в [1] подхода состоит в реализации следующих решений:

– обрабатывается плотное множество 3D точек, получаемых, как за счет данных от 3D сканера (или его виртуального аналога), так и за счет сопоставления особенностей на последовательных во времени снимках стереопар (в начальной версии использовалось только множество, построенное на сопоставленных особенностях);

– в дополнение к ранее разработанному алгоритму формирования 3D точечных моделей ДО на основе критерия “жесткости” предлагается алгоритм “затравочного” типа с использованием критерия “связности точек”;

– задача идентификации и вычисления траектории движения решается также и применительно к ДО, изменяющим свою форму при движении.

Для восстановления движения камеры и объектов сцены применяется метод визуальной навигации (авторские реализации его модификаций описаны в [14–16]). Метод следует классической схеме визуальной одометрии:

– установление соответствий 2D точечных особенностей на двух парах снимков, соответствующим двум соседним расчетным позициям на траектории. Для поиска соответствий точечных особенностей на левом и правом снимке стереопары используется широко применяемый детектор SURF (Speeded Up Robust Features) [17]. А для отслеживания точечных особенностей на двух снимках, соответствующих двум соседним расчетным позициям на траектории, применяется известный трекер KLT (Kanade–Lucas–Tomasi feature tracker) [18];

– построение методом триангуляции лучей по сопоставленному на стереопаре текущей позиции множеству точечных особенностей соответствующего множества 3D точек (3D облако). Аналогичным образом строится 3D облако для стереопары предшествующей позиции. Каждое 3D облако описывается в локальной системе координат (СК) позиции (рис. 1);

– вычисление по сопоставленным 3D облакам матрицы геометрического преобразования, опи-

сывающего относительное перемещение камеры/робота (6 DOF (degrees of freedom)) в локальной СК.

Плотное 3D точечное представление видимой части сцены в каждый момент времени формируется, как было отмечено выше, за счет объединения двух множеств 3D точек, принадлежащих объектам, – точек, получаемых обработкой изображений стереопар (3D облако), и 3D точек, непосредственно измеряемых 3D сканером, или его виртуальным аналогом. Заметим, что первому множеству 3D точек соответствует сопоставленное множество в предшествующей позиции (построенное по сопоставленным особенностям). Это соответствие используется в алгоритме установления связи/соответствия между ДО соседних позиций. Реализуемое плотное точечное представление дает возможность наряду с другими методами выделения объектов в видеопотоке применять и метод, основанный на критерии “связности”, учитывающим непрерывность поверхности объекта.

3. ПОСТРОЕНИЕ ТОЧЕЧНОЙ МОДЕЛИ ДИНАМИЧЕСКОГО ОБЪЕКТА

Задача построения 3D точечных моделей статических и динамических объектов сцены решается на исходном неструктурированном множестве 3D точек (видимых в данной позиции камеры). Все множество 3D точек в текущей позиции i , в которой должны быть сформированы точечные модели статических и динамических объектов, состоит из трех независимо формируемых 3D множеств (см. рис. 2). Здесь:

G_{i-1} – множество 2D особенностей, сопоставленных на изображениях стереопары позиции $(i-1)$ (на предыдущем шаге). $G_{i-1,i}$ множество 2D особенностей в позиции i , прослеженных от G_{i-1} . G_i – множество 2D особенностей, сопоставленных на изображениях стереопары позиции i (на текущем шаге). $G_{i,i+1}$ – множество 2D особенностей в позиции $i+1$, прослеженных от G_i . $M^{i-1}(G_{i-1})$ – множество 3D точек, построенное в СК позиции $(i-1)$ по особенностям G_{i-1} (сформировано на предыдущем шаге). $M_{i-1,i}^i(G_{i-1,i})$ – множество 3D точек, построенное в СК позиции i по особенностям $G_{i-1,i}$. 3D точкам этого множества сопоставлены точки множества $M^{i-1}(G_{i-1})$ в силу выполняемого на текущем шаге прослеживания/сопоставления соответствующих 2D особенностей. $M^i(G_i)$ – множество 3D точек, построенное на текущем шаге в СК позиции i по 2D особенностям G_i . $M_{i,i+1}^{i+1}(G_{i,i+1})$ – множество 3D точек, которое должно быть построено на следующем шаге в СК

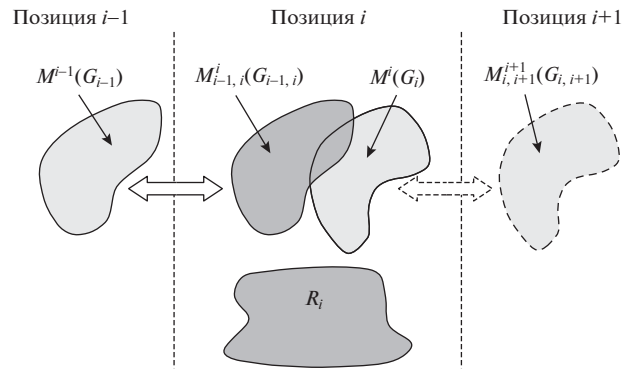


Рис. 2. Объединенное множество исходных 3D точек на текущем шаге: $M_{i-1,i}^i + R_i + M^i$.

позиции $(i+1)$ по особенностям $G_{i,i+1}$. 3D точкам этого множества будут сопоставлены точки множества $M^i(G_i)$ в силу прослеживания/сопоставления соответствующих 2D особенностей на шаге $(i+1)$. R_i – плотное множество 3D точек, полученное “дальномером” на текущем шаге в позиции i .

Под СК позиции понимается система координат камеры съемки, или, что эквивалентно, СК, связанная с роботом, на котором камера жестко закреплена.

Заметим, что сопоставленные 3D множества точек в соседних позициях являются разными координатными представлениями одного и того же множества точек реальных объектов, видимых камерой в разные моменты времени (из двух соседних позиций).

Следует также отметить, что согласно предлагаемой схеме обработки, формируемое на текущем шаге множество 2D особенностей G_i сохраняется, и используется двояким образом. Во-первых, на текущем шаге по сопоставленным на стереопаре особенностям строится множество 3D точек $M^i(G_i)$. А, во-вторых, на следующем шаге (в позиции $(i+1)$) эти особенности прослеживаются с помощью KLT и используются для построения множества 3D точек $M_{i,i+1}^{i+1}$, которое будет сопоставлено с M^i . Таким образом, устанавливается связь позиций i и $(i+1)$, что необходимо для идентификации ДО на протяжении всего времени их движения в поле зрения камеры.

Построения 3D точечных моделей статических и динамических объектов сцены выполняется в несколько этапов. Вначале генерируются т.н. “затравочные” точки, каждая из которых принадлежит какому-то объекту. При этом, возможно, какие-то “затравочные” точки могут принадлежать одному и тому же объекту. Затем выполняется сортировка точек исходного множества по

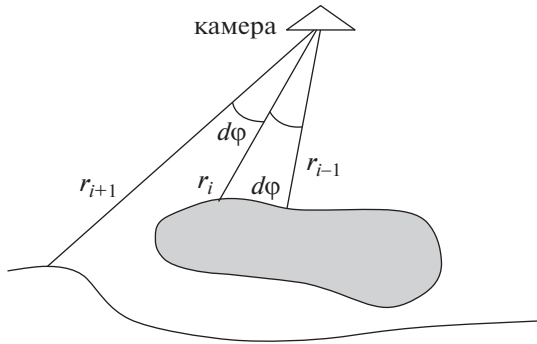


Рис. 3. Проверка принадлежности 3D точки объекту по критерию связности.

группам (отдельным подмножествам точек), порожденным “затравочными” точками, т.е. генерация 3D точечных моделей объектов. Отбор точек осуществляется применением двух критериев, – критерия “связности” и критерия “жесткости”, о которых говорится ниже. Тот факт, что одному объекту могут принадлежать несколько “затравочных” точек, и, соответственно, несколько отдельных групп точек, не противоречит этой методике, поскольку такие группы объединяются в единую 3D модель объекта на основе критерия “сходства движения”. Поскольку сформированные группы точек могут относиться, как к статическим, так и к динамическим объектам, в последующем изложении будем использовать понятие динамической группы (ДГ), являющейся точечным представлением ДО или его части.

3.1. Алгоритм генерации затравочных точек

В простом для реализации варианте затравочные точки могут выбираться на регулярной пространственной решетке, ограничивающей сцену. Этого достаточно с учетом вышесказанного о возможности объединения групп точек, принадлежащих одному объекту.

3.2. Алгоритм формирования множеств точек по критерию жесткости

Критерий “жесткости” строим исходя из двух положений: а) расстояние между двумя точками недеформируемого/жесткого объекта сохраняется при его движении; б) в качестве точки заведомо принадлежащей конкретному объекту будем рассматривать его “затравочную” точку. Тогда отбор тех точек из исходного множества, которые принадлежат данному объекту, можно осуществить с помощью следующего критерия:

пусть $P_{\text{затр}}$ – затравочная точка, а P тестируемая точка в первом 3D облаке,

$P'_{\text{затр}}$ и P' их сопоставленные образы во втором 3D облаке, а

$$r1 = P_{\text{затр}} - P, \quad r2 = P'_{\text{затр}} - P'.$$

Если $r1 = r2$ (с определенной пороговой погрешностью), то точка P принадлежит объекту, представленному затравочной точкой $P_{\text{затр}}$. Чтобы исключить использование порога, реализуется другая форма критерия: выбирается $\min|r1 - r2|$ после вычисления этой разности для всех “затравочных” точек.

3.3. Алгоритм формирования групп точек по критерию связности

Согласно алгоритму “затравочного” типа обработка точек исходного множества 3D точек начинается с затравочной точки (заведомо принадлежащей объекту). Для каждой пары соседних точек (очередная анализируемая точка и предшествующая точка, получившая статус принадлежности к группе) проверяется выполнение критерия “связности”. Согласно критерию, сравниваются расстояния от камеры до текущей и до предшествующей соседней точки (рис. 3):

Пусть $\varepsilon(r_k, r_{k-1}) = |dr/d\varphi|$, где $d\varphi$ – смещение по углу между лучами для двух соседних точек, а $|dr| = |r_k - r_{k-1}|$. Тогда величина $\text{coherence} = \varepsilon2(r_{i+1}, r_i)/\varepsilon1(r_i, r_{i-1})$ будет характеризовать связность/непрерывность поверхности. Если $\text{coherence} >$ пороговой величины, то имеет место разрыв поверхности, т.е. выход на границу объекта. В противном случае текущая точка принадлежит группе (объекту).

3.4. Алгоритм объединения групп точек со сходным движением

Сформированные алгоритмом связности группы точек в текущей позиции i могут относиться не только к разным объектам, но и к одному и тому же объекту. Поэтому для такого рода групп необходимо решать задачу их объединения в одну группу. Это важно для решения задачи обнаружения точек всех статических объектов в сцене и для восстановления движения каждого из ДО.

Исходим из предположения, что в каждой группе присутствуют точки множества R_i и множества $M_{i-1,i}^i$. Алгоритм объединения основывается на применении критерия “жесткости”, который позволяет оценить сходство движения точек. Заметим, что алгоритм применим только к сопоставленным в двух позициях точкам (точки множества R_i не сопоставлены). Обозначим множество сформированных на текущем шаге групп через G . Для каждой группы $g_k \in G$ ищем на множестве $\{G - g_k\}$ группы со схожим движением, применяя указанный критерий. Пусть группа $g_l \in G$ проверяется на предмет объединения с группой g_k . Тогда

согласно критерию “жесткости” рассматриваются две пары точек (рис. 4). В качестве первой точки в первой паре берется точка P^{g_k} в группе g_k , принадлежащая множеству $M_{i-1,j}^i$. В качестве второй точки берется точка P^{g_l} в группе g_l , также принадлежащая множеству $M_{i-1,j}^i$. Во второй паре в качестве первой точки берется точка P^{g_k} , являющаяся прообразом точки P^{g_k} . А в качестве второй точки берется соответственно точка P^{g_l} , являющаяся прообразом точки P^{g_l} . Обе точки принадлежат множеству $M_{i-1,i}^{i-1}$, заданному в СК позиции $(i-1)$. Построим критерий “жесткости”:

$$r1 = |P^{g_k} - P^{g_l}|, \quad r2 = |P^{g_k} - P^{g_l}|.$$

Если $r1 = r2$ (с определенной пороговой погрешностью) для нескольких точек P^{g_l} , то считаем, что обе группы принадлежат “статике” или одному ДО, и, соответственно, могут быть объединены.

Если в группе g_k отсутствуют точки множества $M_{i-1,i}^i$, то считаем, что эта группа точек принадлежит новому ДО. Таким образом, используя сопоставленные пары 3D облаков предыдущей и текущей позиции, определяем одинаковые группы со схожими движениями. После работы “алгоритма объединения” должны остаться только группы, каждая из которых полностью представляет ДО.

4. СОПОСТАВЛЕНИЕ ДИНАМИЧЕСКИХ ГРУПП ТОЧЕК СОСЕДНИХ ПОЗИЦИЙ

Поскольку формирование динамических групп (ДГ) выполняется независимо при обработке каждой позиции траектории камеры, необходимо решать задачу идентификации каждой группы по отношению к группам предыдущей позиции. Т.е. необходимо устанавливать соответствие двух ДГ, фиксируемых/наблюдаемых в разные моменты времени, но представляющих один и тот же объект. Это требуется для расчета траектории движения каждого объекта на протяжении всего времени и для построения его анимационной 3D модели. Трудность заключается в том, что в каждой позиции формирование ДГ точек, принадлежащих ДО, выполняется безотносительно к предыдущей позиции. Дополнительная трудность – непредсказуемое появление новых ДО и исчезновение из сцены прежних ДО, а также необходимость различать недеформируемые и деформируемые ДО. В рассматриваемых ниже алгоритмах установления соответствия между ДГ соседних позиций используется локальная матрица геометрического преобразования, описывающая относительное совместное движение объекта и

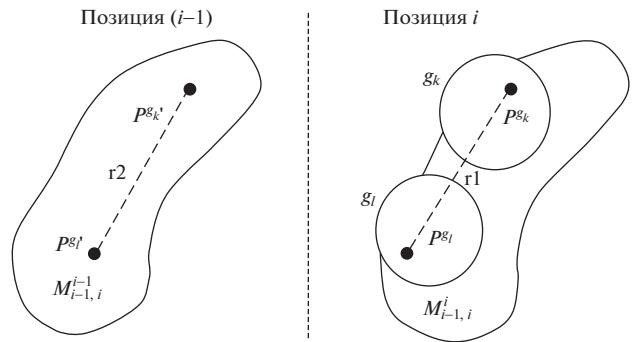


Рис. 4. Объединение группы g_l с группой g_k по критерию “жесткости”.

камеры. Здесь следует заметить, что вычисление этой матрицы делается по-разному для недеформируемых и деформируемых ДО.

4.1. Вычисление локальной матрицы преобразования (относительное движение) для групп точек, представляющих недеформируемые динамические объекты

Для каждой ДГ, сформированной в позиции i , вычисляем методом визуальной одометрии матрицу преобразования координат $H_{i-1,i}^{joint,id_{gr}}$ точек этой ДГ из СК позиции $(i-1)$ в СК позиции i . Здесь id_{gr} – идентификатор (номер) ДГ. Заметим, что $H_{i-1,i}^{joint,id_{gr}}$ – матрица комплексного преобразования, поскольку она отражает результат одновременного движения камеры и ДО (представленного данной ДГ). Движению объекта относительно камеры можно сопоставить симметричное движение камеры относительно объекта. Поэтому матрицу локального комплексного преобразования (вычисленного по сопоставленным 3D облакам этого объекта в соседних позициях) можно представить как:

$$H_{i-1,i}^{joint,id_{gr}} = H_{i-1,i}^{camera} \cdot H_{i-1,i}^{id_{gr}}, \quad (1)$$

здесь $H_{i-1,i}^{camera}$ – матрица, описывающая движение камеры (предварительно вычислена по множеству точек статических объектов), а $H_{i-1,i}^{id_{gr}}$ – матрица, описывающая собственное относительное перемещение данного ДО.

Следует отметить, что первый способ вычисления матрицы $H_{i-1,i}^{joint,id_{gr}}$ основывается на использовании в качестве исходных данных точек ДГ, которые принадлежат сопоставленным множествам $M_{i-1,i}^{i-1}$ и $M_{i-1,i}^i$ (как это делалось в первой версии системы [1]). Однако при наличии в ДГ точек плотного множества R (полученных дальнометром) возможно использование расширенного ис-

ходного множества точек для более точного вычисления матрицы. В этом случае для вычисления матрицы применяется алгоритм ИСР, работающий на не сопоставленных явным образом исходных множествах точек, поскольку, как это было выше указано, множества R формируются в каждой позиции независимо друг от друга.

4.2. Вычисление локальной матрицы преобразования для групп точек, представляющих деформируемые динамические объекты

В случае ДО с изменяемой во времени формой (деформируемых), его точки могут осуществлять разнонаправленные собственные движения (в отличие от недеформируемых). Соответственно, мы не можем вычислить локальную матрицу (которая объединяет движение камеры и собственное движение объекта) $H_{i-1,i}^{joint,id_{gr}}$ методом визуальной одометрии, как это делается для недеформируемых объектов. Поэтому будем рассматривать/моделировать движение деформируемого ДО, как движение его центра масс (далее будем называть центральной точкой), определяемого по множеству принадлежащих данной ДГ точек, сопоставленных на шаге $[i-1, i]$. Тогда относительное перемещение центральной точки ДГ из позиции $(i-1)$ в позицию i можно описывать матрицей, объединяющей матрицу движения камеры и матрицу переноса (описывающую собственное перемещение центральной точки). Но, в данном случае (в отличие от вышеописанного случая недеформируемых ДО) матрица, описывающая собственное движение ДО, содержит только вектор перемещения центра масс (a, b, c):

$$H_{i-1,i}^{id_{gr}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ a & b & c & 1 \end{pmatrix} - \text{матрица, описывающая собственное перемещение центральной точки на неизвестный вектор (a, b, c) за время перемещения камеры из позиции (i-1) в позицию i.}$$

Пусть:

$P_{center}^{CS^{i-1},id_{gr}}$ – центральная точка ДГ с именем id_{gr} , измеренная в $СК^{i-1}$ (по множеству сопоставленных точек);

$P_{center}^{CS^{i-1},id_{gr},matched} = P_{center}^{CS^{i-1},id_{gr}} \cdot H_{i-1,i}^{camera}$ – образ центральной точки $P_{center}^{CS^{i-1},id_{gr}}$ в $СК^i$, полученный с учетом перемещения камеры (без учета собственного движения);

$P_{center}^{CS^i,id_{gr}}$ – центральная точка, измеренная в $СК^i$ (по множеству сопоставленных точек).

Тогда элементы a, b, c в матрице $H_{i-1,i}^{id_{gr}}$ определяем как компоненты вектора смещения ΔP (a, b, c) = $P_{center}^{CS^i,id_{gr}} - P_{center}^{CS^{i-1},id_{gr},matched}$

Таким образом, мы сформировали комплексную матрицу

$$H_{i-1,i}^{joint,id_{gr}} = H_{i-1,i}^{camera} \cdot H_{i-1,i}^{id_{gr}},$$

связывающую координаты центральной точки ДГ (деформируемого ДО) в $СК^{i-1}$ и в $СК^i$.

4.3. Алгоритм идентификации динамических групп точек в текущей позиции

Как было отмечено выше, для прослеживания всех ДГ на протяжении всего времени наблюдения/съемки необходимо идентифицировать каждую из ДГ на каждом очередном шаге. Это означает, что для сформированной в новой позиции ДГ требуется установить ее соответствие с одной из ДГ, сформированной в предшествующей позиции. Поскольку сформированные в каждой позиции ДГ уже разделены на два типа – недеформируемые и деформируемые группы, то поиск соответствия для сформированной в новой позиции недеформируемой ДГ осуществляется на множестве ДГ этого же типа предшествующей позиции. То же самое справедливо и для деформируемых ДГ. Такое соответствие для недеформируемой ДГ, сформированной в позиции (i) , может быть установлено двумя способами. Первый способ – соответствие устанавливается алгоритмом, основанным на сравнении вычисляемых интегральных оценок геометрической близости точек этой ДГ к пространственным оболочкам ДГ в позиции $(i-1)$. Второй способ – соответствие устанавливается проверкой наличия сопоставленных 3D точек в двух анализируемых ДГ. Недостаток алгоритма оценки геометрической близости – вычислительная трудоемкость. Поэтому был выбран способ решения, основанный на выявлении в сравниваемых ДГ сопоставленных 3D точек, которые служат “маркерами”. Наличие таких точек в обеих группах, принадлежащих, соответственно, сопоставленным множествам $M_{i-1,i}^{i-1}$ и $M_{i-1,i}^i$, подтверждает, что обе точечные модели представляют один и тот же ДО. Поэтому алгоритм идентификации некоторой ДГ в текущей позиции i по отношению к предыдущей позиции состоит в последовательной проверке всех ДГ в предыдущей позиции на наличие в обеих группах общих сопоставленных точек, принадлежащих указанным множествам. Иллюстрация к работе алгоритма показана на рис. 5. В случае успешного сопоставления тестируемой ДГ в позиции i ей присваивается идентификатор ее прообраза в позиции $(i-1)$. Одновременно по принципу унаследования свойств определяется и тип ДО – “недеформиру-

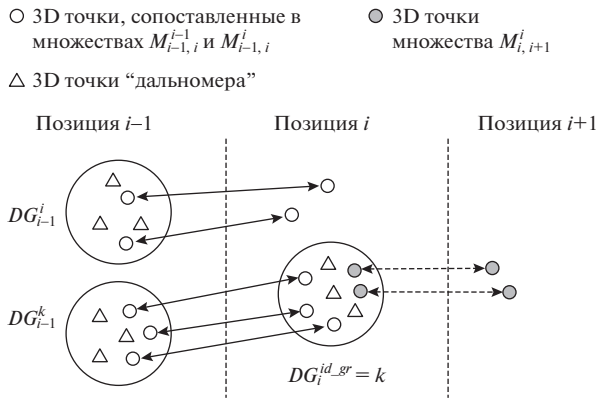


Рис. 5. Идентификация ДГ. Группа точек ДО $DG_i^{id_{gr}}$, сформированная в позиции i , сопоставлена с DG_{i-1}^k в позиции $(i - 1)$ и не сопоставлена с DG_{i-1}^i .

емый” или “деформируемый”. Если такого сопоставления не обнаружено, то это значит, что ДГ относится к новому ДО. Заметим, что возможно применение обоих рассмотренных критериев (алгоритмов) для повышения достоверности идентификации ДГ.

Для нового ДО необходимо определить его тип. Определение типа делается алгоритмом, основанным на применении к точкам сформированной ДГ критерия “жесткости”. Точки ДГ проверяются критерием на сходство движения. Для проверки используются точки, принадлежащие сопоставленным множествам $M_{i-1,i}^{i-1}$ и $M_{i-1,i}^i$.

При сходстве движения точек квалифицируем ДО как “недеформируемый”, в противном случае – как “деформируемый”.

5. О ВЫЧИСЛЕНИИ ТРАЕКТОРИЙ И РЕКОНСТРУКЦИИ ДИНАМИЧЕСКИХ ОБЪЕКТОВ

Каждый ДО характеризуется своим временем жизни в сцене от момента времени его появления в сцене t_b до момента времени выхода его из сцены t_e . С моментом времени t_b связана начальная система координат данного объекта. В этой СК будет описываться собственное движение объекта, безотносительно к движению камеры. Также в этой СК будет выполняться генерация полной 3D модели этого объекта – т.е. в нее будут помещены все полученные виды объекта с последующим объединением. В качестве такой начальной СК объекта выберем (без ущерба общности) локальную СК камеры/робота в позиции b . Координаты точек ДО из его начальной позиции b (а также из других позиций) могут быть преобразованы согласно методу визуальной одометрии в СК на-

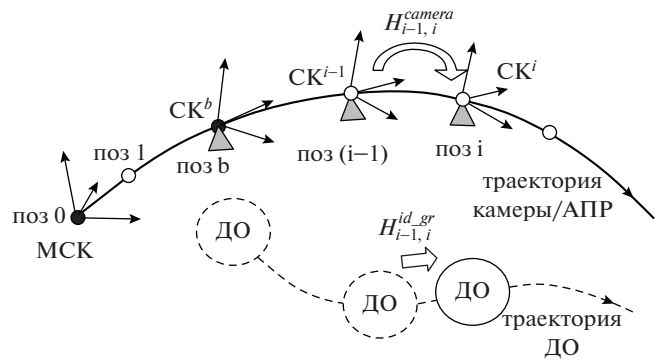


Рис. 6. Вычисление матриц геометрического преобразования, определяющих собственное движение динамического объекта. Позиция b – начало движения ДО в сцене.

чальной позиции траектории робота (t_0). В свою очередь, при наличии преобразования, связывающего СК начальной позиции траектории робота с некоторой внешней системой координат, можно получить координаты объектов сцены во внешней системе координат. Назовем эту внешнюю СК мировой системой координат (МСК). В качестве МСК часто используют (без ущерба общности) СК робота в начальной позиции, предполагая, что упомянутое выше преобразование во внешнюю СК известно. Собственное относительное перемещение ДО за время $[t_{i-1}, t_i]$ будем оценивать вычислением локальной матрицы геометрического преобразования на шаге $[(i - 1), i]$ (см. (1)).

Вычисление траектории ДО. В качестве траектории ДО можно рассматривать траекторию движения любой его точки из начальной позиции b (момент времени t_b) в конечную позицию e (момент времени t_e). Построить такую траекторию можно вычисляя последовательно локальные матрицы геометрического преобразования $H_{i-1,i}^{id_{gr}}$ (рис. 6), которые определяют собственное движение ДО в интервалах времени $t_b - t_1, t_1 - t_2, \dots, t_{i-1} - t_i, \dots, t_{e-1} - t_e$.

Из (1) следует:

$$H_{i-1,i}^{id_{gr}} = H_{i-1,i}^{joint,id_{gr}} \cdot (H_{i-1,i}^{camera})^{-1} \quad (2)$$

То есть, по вычисленному движению камеры ($H_{i-1,i}^{camera}$) и совместному движению объекта и камеры ($H_{i-1,i}^{joint,id_{gr}}$) мы можем вычислить матрицу геометрического преобразования координат $H_{i-1,i}^{id_{gr}}$, определяющую собственное локальное перемещение ДО. Тогда преобразование 3D координат точки $P_i^{id_{gr}}$ объекта id_{gr} на шаге $[(i - 1), i]$ определяется соотношением:

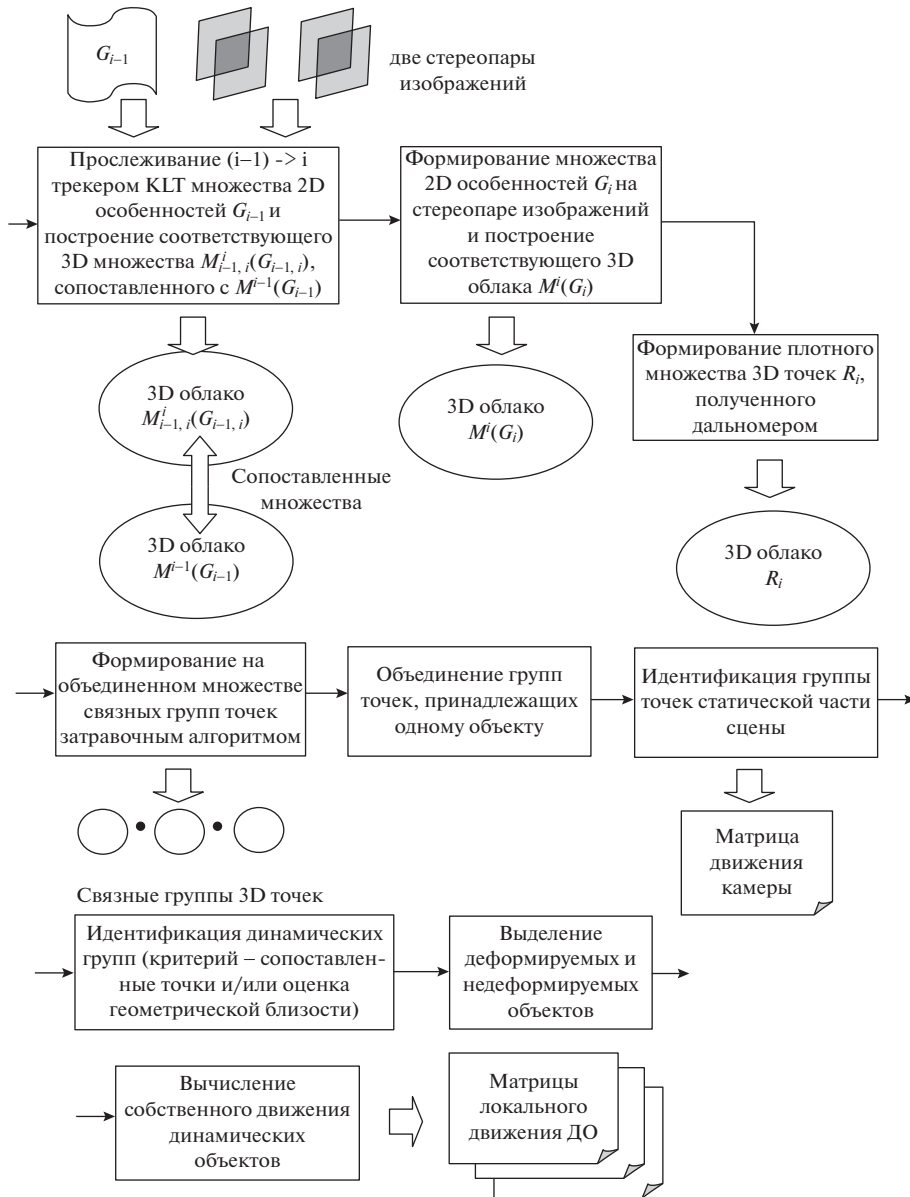


Рис. 7. Вычислительная схема обработки данных на текущем шаге траектории.

$$P_l^{t_i, id_{gr}} = P_l^{t_{i-1}, id_{gr}} \cdot H_{i-1, i}^{id_{gr}}, \quad (3)$$

здесь $P_l^{t_i, id_{gr}}$ – координаты точки l объекта id_{gr} в момент времени t_i .

Матрица результирующего преобразования ДО с именем id_{gr} за время $[t_b, t_i]$ (собственное относительное движение) получается перемножением соответствующих локальных матриц:

$$H_{b, i}^{id_{gr}} = \prod_{k=b}^{i-1} H_{k, k+1}^{id_{gr}}, \quad (4)$$

где b – позиция камеры в момент появления ДО с именем id_{gr} .

С учетом (4) координаты точек объекта id_{gr} в момент времени t_i связаны с координатами точек в начальной СК этого объекта матрицей $H_{b, i}^{id_{gr}}$:

$$P_l^{t_i, id_{gr}} = P_l^{t_b, id_{gr}} \cdot H_{b, i}^{id_{gr}} \quad (5)$$

Для помещения всех ДО и траекторий их движения в единую СК, которой является МСК, необходимо иметь преобразование из начальной СК ДО в МСК. Поскольку в качестве МСК мы рассматриваем локальную систему координат камеры в началь-

ный момент времени, то $H_{WCS, b}^{camera} = \prod_{k=0}^{b-1} H_{k, k+1}^{camera}$. Со-

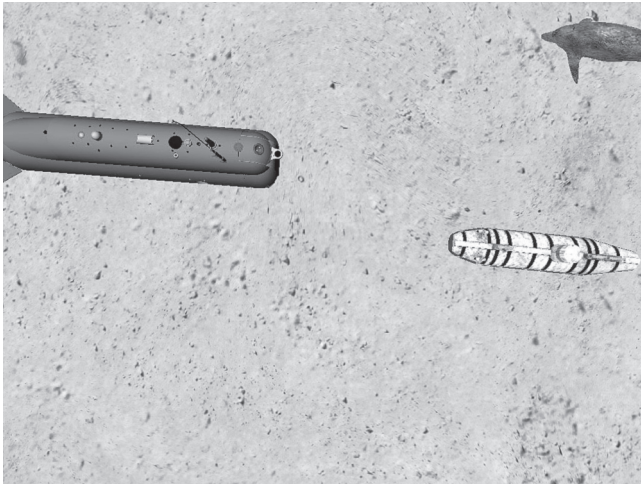


Рис. 8. Снимок подводной сцены в одной из позиций траектории АПР: в поле зрения камеры находятся 3 динамических объекта.

ответственно, координаты рассматриваемой точки $P_l^{t_i, id_{gr}}$ в МСК могут быть получены как:

$$P_l^{WCS, id_{gr}} = P_l^{t_i, id_{gr}} \cdot (H_{WCS, b}^{camera})^{-1}, \quad (6)$$

3D реконструкция ДО. Чтобы построить 3D модель объекта по видам объекта, полученным из разных позиций камеры (в разные моменты времени), необходимо совместить эти виды в СК начальной позиции этого ДО. Это реализуется с помощью следующих операций:

1) Из каждой позиции i траектории камеры видимые точки $P_n^{CK^i, id_{gr}}$ объекта id_{gr} , построенные по

стереопаре в СК ^{i} , помещаются в СК начальной позиции этого ДО, т.е. в СК позиции b камеры (момент появления в сцене этого ДО). Это делается с помощью матрицы $H_{b, i}^{joint, id_{gr}}$, которая учитывает и движение камеры, и преобразование координат точек ДО, связанное с его перемещением:

$$P_n^{CK^{i, b}, id_{gr}} = P_n^{CK^i, id_{gr}} \cdot (H_{b, i}^{joint, id_{gr}})^{-1} \text{ здесь}$$

$P_n^{CK^{i, b}, id_{gr}}$ – 3D точка с номером n объекта id_{gr} , преобразованная из СК ^{i} в СК ^{b} .

Матрица $H_{b, i}^{joint, id_{gr}}$ согласно методу визуальной одометрии получается перемножением соответствующих локальных матриц:

$$H_{b, i}^{joint, id_{gr}} = \prod_{k=b}^{k=i-1} H_{k, k+1}^{joint, id_{gr}}, \text{ где локальная матрица}$$

$H_{k, k+1}^{joint, id_{gr}}$ определяет совместное движение камеры и ДО из позиции k в позицию $(k + 1)$.

2) Чтобы получить координаты точек объектов всех видов (ракурсов) в МСК необходимо применить преобразование $(H_{WCS, b}^{camera})^{-1}$, где $H_{WCS, b}^{camera} = \prod_{k=0}^{k=b-1} H_{k, k+1}^{camera}$.

Таким образом, возможность пересчета координат точек каждого ДО в МСК позволяет реконструировать их траектории и 3D модели в едином координатном пространстве мировой системы координат.

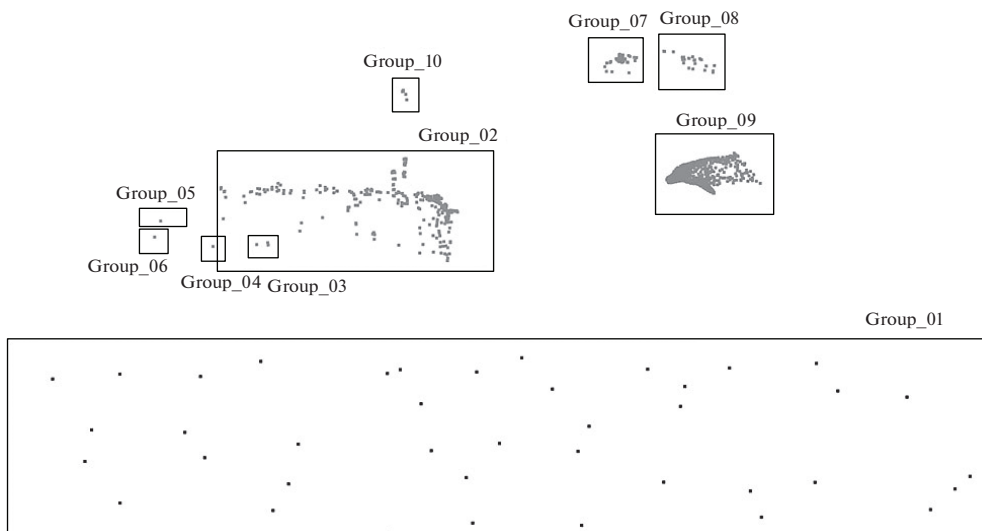


Рис. 9. Показана проекция “вид сбоку”. Каждая из 9 сформированных групп 3D точек выделена прямоугольником.

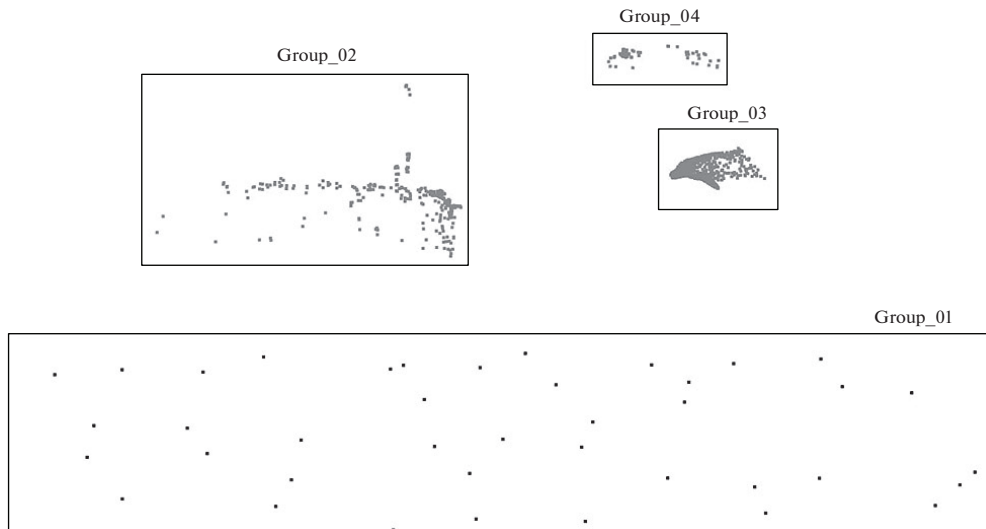


Рис. 10. Выделены 3 группы точек, относящихся к ДО и группа точек, представляющих рельеф дна (после работы алгоритма объединения).

6. СХЕМА ОБРАБОТКИ ДАННЫХ В ЗАДАЧЕ ИДЕНТИФИКАЦИИ И ВОССТАНОВЛЕНИЯ ДВИЖЕНИЯ ДИНАМИЧЕСКИХ ОБЪЕКТОВ ПО ИЗОБРАЖЕНИЯМ

Обработка данных с вычислением траектории движения камеры/робота и ДО осуществляется пошагово. На каждом шаге (в текущей позиции) используется метод визуальной одометрии и выполняется формирование и идентификация точечных моделей статических и динамических объектов с помощью вышеописанной алгоритмической базы.

Вычислительная схема обработки данных на текущем шаге траектории, согласно предлагаемому подходу, состоит из выполнения следующих этапов (рис. 7):

1. Прослеживание с помощью трекера KLT 2D множества особенностей G_{i-1} , сформированного в позиции $(i - 1)$ на предыдущем шаге. Построение соответствующего 3D множества $M_{i-1,i}^i$, сопоставленного с M^{i-1} .

2. Формирование на стереопаре изображений множества 2D особенностей G_i и построение соответствующего 3D облака точек M^i .

3. Формирование “дальномерного” плотного множества 3D точек R_i .

4. Формирование на объединенном множестве $\{M_{i-1,i}^i + R_i + M^i\}$ связанных групп точек затравочным алгоритмом (на критерии связности). Группы объединяют точки видимых поверхностей разных объектов или разных частей одного и того же объекта.

Отметим, что в объединенном множестве каждая точка сохраняет статус своей принадлежности к одному из трех указанных множеств, т.к. это необходимо для работы алгоритмов идентификации ДО и вычисления матриц их движения.

5. Объединение в одну группу тех групп точек, которые принадлежат одному и тому же объекту. Объединение выполняется алгоритмом, оценивающим сходство движения.

6. Идентификация группы точек статических объектов. Вычисление методом визуальной одометрии матрицы локального перемещения камеры по сопоставленным точкам статической части сцены (принадлежащим множествам M^{i-1} и $M_{i-1,i}^i$).

Исключая из всех полученных групп группу “статика”, получаем ДГ, т.е. группы точек, принадлежащие ДО (недеформируемым и деформируемым).

7. Идентификация ДГ по отношению к ДГ предыдущей позиции.

8. Разделение деформируемых и недеформируемых ДО.

9. Вычисление собственного движения ДО.

Обработка первых двух позиций траектории отличается от приведенной схемы тем, что для позиции 1 отсутствует предыдущий шаг, и, соответственно сегментация точек по группам выполняется на множествах $R_1 + M_{1,2}^1$.

7. ВЫЧИСЛИТЕЛЬНЫЕ ЭКСПЕРИМЕНТЫ

Тестирование описанных алгоритмов осуществлялось на подводных виртуальных сценах, в которых съемка ведется с автономного подводного робота (АПР). Пример одной из сцен показан

Таблица 1. Оценка эффективности алгоритмов идентификации ДО

Объект	Кол-во кадров, на которых был определен объект	Усредненная локальная погрешность между соседними кадрами, мм	Накопленная погрешность, мм
подводный аппарат	55	0.4	12.6
батискаф	26	0.6	7.2
дельфин	64	0.6	15.3

Таблица 2. Оценка эффективности всей вычислительной схемы с учетом этапа сопоставления особенностей на стереопаре снимков

Объект	Кол-во кадров, на которых был определен объект	Усредненная локальная ошибка между соседними кадрами, мм	Накопленная ошибка, мм
подводный аппарат	55	0.7	40.4
батискаф	26	0.7	24.8
дельфин	64	0.8	53.1

на рис. 8. В сцене присутствуют три ДО (“подводный аппарат”, “батискаф” и “дельфин”). Алгоритм формирования групп точек по критерию связности сформировал 9 групп 3D точек для данного вида (рис. 9). После работы алгоритма объединения групп точек со сходным движением сформированы 3 группы точек, принадлежащие 3-м указанным динамическим объектам и группа точек, относящаяся к рельефу дна (рис. 10). Данные объекты были прослежены с шагом = 1 кадр на сцене протяженностью в 120 кадров, длительностью 12 сек. Робот/камера и все ДО двигались по прямолинейным траекториям с разными скоростями под разными углами. Время наблюдения объектов (пребывания их в сцене) составило от 2.6 сек (26 кадров) до 6.4 сек (64 кадра).

Другая серия экспериментов была выполнена для оценки точности локализации ДО. Для некоторых позиций камеры (т.е. для отдельных моментов времени) вычислялась локальная (относительная) точность для каждого ДО (погрешность определения координат за время прохождения между двумя соседними позициями) и абсолютная точность (погрешность определения координат, накопленная к данному моменту времени в некоторой внешней системе координат за время движения от начала траектории). Эксперименты выполнялись в два этапа. На первом этапе оценивалась эффективность разработанных алгоритмов при условии безошибочной работы детектора и трекера по сопоставлению особенностей на снимках (использовалась имитация работы детектора и SURF и трекера KLT). Эти результаты представлены в табл. 1.

На втором этапе оценивалась эффективность всей вычислительной схемы, в том числе, с уче-

том ошибок, вносимых на этапе обработки стереопар снимков (табл. 2).

Как видно из таблиц, полученные результаты подтвердили эффективность предложенных алгоритмов и вычислительной схемы в целом. Этот результат соответствует ранее полученным авторами в [1] оценкам точности локализации.

8. ЗАКЛЮЧЕНИЕ

Рассмотрена модификация ранее предложенного авторами подхода к решению 3D SLAM задачи для динамических сцен, основанного на точечном представлении объектов. Модификация заключается в реализации более эффективной алгоритмической основы, чем это было в начальной версии, а также в реализованных возможностях генерации расширенного точечного представления (за счет применения “виртуального дальномера”) и идентификации динамических объектов с неизменяемой и изменяемой формой.

Расширенное точечное представление позволяет получать более плотные точечные модели ДО. Результаты вычислительных экспериментов на модельных сценах подтвердили правильность выбранных решений. Дальнейшее развитие работы предполагает: совершенствование алгоритмической базы; тестирование программ на разных наборах данных; решение задачи 3D реконструкции ДО с использованием вычисляемых траекторий объектов.

9. БЛАГОДАРНОСТИ

Работа выполнена при частичной финансовой поддержке РФФИ (проект № 18-07-00165).

СПИСОК ЛИТЕРАТУРЫ

1. *Бобков В.А., Кудряшов А.П., Мельман С.В.* О восстановлении движения динамических объектов по стереоизображениям // Программирование. 2018. № 3. С. 29–42 (A. Bobkov, A. P. Kudryashov, S. V. Mel'man. On the Recovery of Motion of Dynamic Objects from Stereo Images // Programming and Computer Software, 2018, Volume 44, Issue 3, pp. 148–158).
2. *Hasler N., Rosenhahn B., Thormahlen T., Wand M., Gall J., Seidel H.P.* Markerless motion capture with unsynchronized moving cameras // Conference on Computer Vision and Pattern Recognition, 2009, pp. 224–231.
3. *Ballan L., Brostow G.J., Puwein J., Pollefeys M.* Unstructured video-based rendering: Interactive exploration of casually captured videos. ACM Transactions on Graphics // Proceedings of SIGGRAPH, 2010. P. 134–146.
4. *Taneja A., Ballan L., Pollefeys M.* Modeling dynamic scenes recorded with freely moving cameras // Conference on Computer Vision, 2011. P. 613–626.
5. *Wedel A., Brox T., Vaudrey T., Rabe C., Franke U., Cremers D.* Stereoscopic scene flow computation for 3D motion understanding // Intl. J. of Computer Vision. 2011. V. 35. № 1. P. 29–51.
6. *Alcantarilla P., Yebes J., Almazn J., Bergasa L.* On combining visual SLAM and dense scene flow to increase the robustness of localization and mapping in dynamic environments // International Conference on Robotics and Automation, May 2012. P. 1290–1297.
7. *Keller M., Lefloch D., Lambers M., Izadi S., Weyrich T., Kolb A.* Real-time 3d reconstruction in dynamic scenes using point-based fusion // Proc. of Joint 3DIM/3DPVT Conference (3DV). 2013. P. 1–8.
8. *Mustafa A., Kim H., Guillemaut J.-Y., Hilton A.* General Dynamic Scene Reconstruction from Multiple View Video // International Conference on Computer Vision. 2015. P. 900–908.
9. *Mustafa A., Kim H., Guillemaut J.-Y., Hilton A.* Temporally coherent 4D reconstruction of complex dynamic scenes // IEEE Conference on Computer Vision and Pattern recognition. 2016. P. 223–245.
10. *Lefloch D., Kluge M., Sarbolandi H., Weyrich T., Kolb A.* Comprehensive Use of Curvature For Robust And Accurate Online Surface Reconstruction // IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, Vol. 39, № 12, p. 2349–2365.
11. *Камаев А.Н., Сухенко В.А., Карманов Д.А.* Построение и визуализация трехмерных моделей морского дна для тестирования систем технического зрения АНПА // Программирование. 2017. № 3. С. 69–82.
12. *Золотов В.А., Петрищев К.С., Семенов В.А.* Исследование методов пространственного индексирования динамических сцен на основе регулярных октодеревьев // Программирование. 2016. № 6. С. 59–66.
13. *Bobkov V.A., Ronshin Y.I.* GPU Implementation of Depth Map Algorithm // The First Russia and Pacific Conference on Computer Technology and Applications, Vladivostok, 6–9 September, 2010, p. 382–387.
14. *Бобков В.А., Роньшин Ю.И., Кудряшов А.П., Машенцев В.Ю.* 3D SLAM по стереоизображениям // Программирование. 2014. № 4. С. 5–12. (V. A. Bobkov, Yu. I. Ron'shin, A. P. Kudryashov, and V. Yu. Mashentsev. 3D SLAM from Stereoimages // Programming and Computer Software, 2014. Vol. 40, No. 4, 2014, pp. 159–165).
15. *Bobkov V.A., Mel'man S.V., Kudryashov A.P.* Fast computation of local displacement by stereo pairs // Pattern Recognition and Image Analysis, 2017. V. 27. № 3. P. 458–465.
16. *Bobkov V.A., Kudryashov A.P., Mel'man S.V., Shcherbatyuk A.F.* Autonomous Underwater Navigation with 3D Environment Modeling Using Stereo Images // Gyroscopy and Navigation, 2018. V. 9. № 1. P. 67–75.
17. *Bay H., Ess A., Tuytelaars T., Van Gool L.* Speeded-Up Robust Features (SURF) // Computer Vision and Image Understanding, 2008. V. 110. P. 346–359.
18. *Lucas B.D., Kanade T.* An Iterative Image Registration Technique with an Application to Stereo Vision // International Joint Conference on Artificial Intelligence, 1981. P. 674–679.