УДК 004.932.2

# А. В. Смирнов, А. Ю. Беззубцев

# Обход препятствий подвижными техническими средствами с использованием стереозрения

Аннотация. В статье предложен метод обхода препятствий на пути подвижного технического средства (ПТС) с использованием алгоритмов стереозрения и системы распределённой блочно параллельной обработки данных. Также в статье описывается разработанный алгоритм генерации карты исследуемого помещения, рассматривается применение алгоритма А\* для расчёта пути обхода, и выдвигается концепция создания реального ПТС для тестирования алгоритмов.

Ключевые слова и фразы: Подвижно техническое средство(ПТС), карта глубины, сетереозрение, ректификация, А\*, распределенная система, Raspberry Pi.

#### Введение

Одной из проблем автоматической навигации роботизированных подвижных технических средств (далее ПТС) является обход встречающихся препятствий. Задача обхода препятствий не тривиальна и не смотря на общее развитие робототехники и сенсоров, её решение до сих поря является актуальным. Значительного прогресса в решении данной задачи можно добиться используя алгоритмы построения пути. Большинство таких алгоритмов основаны на поиске пути на графе. К ним можно отнести алгоритмы поиска в ширину и поиска в глубину [1], а также различные вариации алгоритма А<sup>\*</sup>. В данной статье будет рассмотрена упрощенная реализация алгоритма А<sup>\*</sup>, достаточная для решения поставленной задачи.

Тем не менее, на практике для применения алгоритмов построения пути необходимо иметь представление о взаимном расположении препятствий и близких к ПТС объектов, то есть составить некую карту наблюдаемой сцены. Определить взаимное расположение объектов можно с помощью анализа данных о глубине сцены. Такие данные могут дать специализированные сканирующие устройства

Работа поддержана грантом Министерством образования и науки Российской Федерации (идентификатор № RFMEFI60714X0012).

<sup>©</sup> А. В. Смирнов, А. Ю. Беззубцев, 2016

<sup>©</sup> Институт программных систем имени А. К. Айламазяна РАН, 2016

<sup>©</sup> Программные системы: теория и приложения, 2016

такие как сканирующий лазерный дальномер (СЛД) или лидар,с использованием методов и алгоритмов SLAM(Simultaneous Localization And Mapping) [2]. Стоит отметить, что SLAM это не какой-то конкретный алгоритм, а набор методов и техник, которые позволяют решать поставленную выше задачу. Однако, не смотря на то, что с помощью SLAM можно создать карту сцены, но для его использования необходимо дорогостоящее оборудование.

Также данные о глубине можно получить, используя стереозрение [3], которое является одним из направлений машинного зрения. Его применение позволяет получить расстоянии до объектов, может использоваться при составлении картины окружающего мира [4]. В робототехнике стереозрение дает возможность управления движением роботов, включая наземные машины и беспилотные летательные аппараты.

В отличие от машинного зрения, основанного на использовании одной камеры, в технологии стереозрения используются, как правило, две камеры, работающие синхронно. Стереозрение в определенной степени позволяет компьютеру получать информацию не только о цвете и яркости объекта, но и о расстоянии до него. Это дает возможность обойтись без использования разнообразных дорогостоящих датчиков, сонаров и лидаров. Таким образом, задачу обхода близлежащих препятствий можно решить, используя стереозрение и алгоритмы поиска пути.

## 1. Создание карты глубины

Карта глубины (depth map) представляет собой черно-белое изображение, на котором интенсивность цвета отражает расстояние до объекта. Такая карта может быть получена с помощью специальной камеры глубины (например, сенсор Kinect, является своего рода такой камерой), а так же может быть построена с использованием стереозрения по стереопаре изображений.

Принцип создания карты глубины достаточно прост: для каждого пикселя левого изображения с координатами  $(X_0, Y_0)$  выполняется поиск пикселя на правом изображении. При этом предполагается, что пиксель на правом изображении должен иметь координаты  $(X_0-d, Y_0)$ , где d — величина, называемая несоответствием/смещением (disparity). Поиск соответствующего пикселя выполняется путем вычисления максимума функции отклика, в качестве которой может выступать, например, корреляция окрестностей пикселей. Обход препятствий ПТС с использованием стереозрения



Рис. 1: А — изображение с левой камеры; В — изображение с правой камеры; С — карта глубины, полученная с виртуальной сцены

Для построения карт глубин используется метод полуглобального сопоставления SGBM (Semi-Global Block-Matching Algorithm) [5]. В его основе лежит поиск такого соответствия между пикселями, которое максимизирует взаимную информацию для каждой пары пикселей. Гладкость полученной карты дальностей достигается путем комбинирования нескольких одномерных ограничений.

## 2. Карта глубины виртуальной сцены

Чтобы узнать приблизительное время работы алгоритма SGBM и точность построения карты глубины, была смоделирована виртуальная сцена в редакторе 3D графики, которая представляла собой комнату с двумя объектами, отдалёнными на разное расстояние от стереопары камер. Далее была сгенерирована карта глубины для представленной выше 3D сцены. Все параметры алгоритма SGBM были получены экспериментальным методом. Результат показан на рис. 1).

Несмотря на приемлемый результат, проведённый эксперимент с виртуальной комнатой нельзя считать полностью достоверным, так как камеры в данном эксперименте являются «идеальными». Такие



(а) левое изображение

(b) правое изображение



(с) до преобразования

(d) после преобразования

Рис. 2: Стереопара изображений и карта глубины

камеры не нуждаются в калибровке и ректификации изображений, к тому же реальная сцена может быть более насыщена объектами, чем смоделированная.

## 3. Карта глубины реальной сцены

После тестирования метода SGBM на виртуальной сцене был проведён эксперимент по применению данного метода к реальной стереопаре снимков. Стереоустановка была размещена на высоте равной высоте ПТС. Перед стереоустановкой были расположены два объекта на расстоянии 80 и 120 см соответственно. Расстояние между камерами (стерео база) было установлено на 10 см. Получившаяся стереопара изображений представлена на рисунках 2a и 2b.

Так как камеры в данном эксперименте не «идеальны», стереопара снимков прошла процедуру ректификации. Ректификация (выравнивание) стереопары — это выполнение некоторых преобразований, при которых и правое, и левое изображение проецируются на плоскость, параллельную базовой линии, то есть линии, соединяющей оптические центры камер. Полноразмерная карта глубины показана на рис. 2с. Помимо искомых объектов, карта также содержит достаточно много шума. С целью избавления от большинства шума, на карте глубины ректифицированная стереопара изображений была изменена таким образом, чтобы в поле зрения стереокамеры входили только искомые объекты. На рис. 2d. показана карта глубины после проведённых преобразований. Такое преобразование позволило снизить время работы алгоритма благодаря меньшему количеству обрабатываемых пикселей.

Имея карту глубины, можно получить данные о расстоянии до наблюдаемых объектов. Такие данные понадобятся для создания карты сцены. Алгоритм построения карты наблюдаемой сцены описан в следующей главе.

#### 4. Алгоритм построения карты сцены

Стоит отметить, что для решения поставленной в статье задачи нет необходимости в построении подробной карты наблюдаемой сцены, достаточно знать расположение близлежащих к ПТС объектов и их приблизительные габариты. Такая карта представляет собой бинарное изображение, где чёрным цветом отображаются зафиксированные объекты, а белым свободные области или неизвестные области.

Алгоритм построения карты сцены состоит из нескольких шагов:

- (1) пороговая бинаризация карты глубины. Бинаризация применяется для выделения более светлых объектов, то есть объектов, наиболее близко расположенных к стереоустановке. Опытным путём было найдено расстояния успешного детектирования объектов, равное 200 см. В зависимости от условий, в которых была получена стереопара изображений, на этом этапе возможно использовать адаптивную бинаризацию. Далее с использованием контурного анализа, для объектов рассчитываются габаритные прямоугольники.
- (2) расчёт расстояния до детектированных объектов. Значения глубины обратно пропорциональны величине смещения пикселей. Зависимость между величиной диспаритета (disparity) и глубиной можно выразить следующим способом:  $Z = \frac{fT}{d}$  где Z значение глубины (расстояние), f фокальная длина камеры, T расстояние между камерами, d значение диспаритета в данном пикселе. Зная области, в которых находятся объекты на карте диспаритетов, можно подсчитать среднее значение диспаритета для объекта, пройдя по всем пикселям, принадлежащим ему.



Рис. 3: Бинарная карта сцены

(3) проецирование найденных объектов на бинарную карту. Изначально создаётся пустая (полностью белая) бинарная карта сцены. Создание карты ведётся с учётом соотношения 1 пиксель = 1 см. Найденные объекты наносятся на карту с учётом удалённости от камер и собственных габаритов. Так как по карте глубины нельзя точно определить толщину объектов, на карту наносятся поверхностные слои найденных объектов.

Результатом работы вышеизложенного алгоритма является бинарная карта сцены: рис. 3.

## 5. Алгоритм поиска пути А\*

Для поиска пути обхода обнаруженных препятствий был выбран алгоритм A\*[6]. A\* — в информатике и математике, алгоритм поиска по первому наилучшему совпадению на графе, который находит маршрут с наименьшей стоимостью от одной вершины (начальной) к другой (целевой, конечной). Порядок обхода вершин определяется эвристической функцией, обычно обозначаемой как F(x).

Этот алгоритм был впервые описан в 1968 году Питером Хартом, Нильсом Нильсоном и Бертрамом Рафаэлем. Это по сути было расширение алгоритма Дейкстры, созданного в 1959 году. Новый алгоритм достигал более высокой производительности (по времени) с помощью эвристики.

Принцип работы алгоритма:

- (1) создается 2 списка вершин ожидающие рассмотрения и уже рассмотренные. В ожидающие добавляется точка старта, список рассмотренных пока пуст.
- (2) для каждой вершины рассчитывается стоимость пути, определяющаяся функцией *F*. Так же каждая вершина хранит ссылку на родительскую вершину, из которой в нее пришли.
- (3) из списка вершин на рассмотрение выбирается точка с наименьшей стоимостью. Обозначим ее X.

#### Обход препятствий ПТС с использованием стереозрения



Рис. 4: Найденный на бинарной карте путь от точки старта вверх к цели.

- (4) если X цель, то считается, что путь найден.
- (5) перенос X из списка ожидающих рассмотрения в список уже рассмотренных.
- (6) для каждой из вершин, соседних для X (обозначим эту соседнюю точку Y), делается следующее:
  - если *Y* уже находится в списке рассмотренных вершин, то данная вершина пропускается.
  - если Y еще нет в списке вершин ожидающих рассмотрение, то данная вершина добавляется в список с ссылкой на родительскую вершину X и рассчитанной стоимостью.
  - из списка вершин ожидающих рассмотрения выбирается та, у которой наименьшее стоимость. Данная вершина Y переносится в список рассмотренных вершин и заменяет собой вершину X.
  - если список точек на рассмотрение пуст, а цель не была достигнута — значит маршрут не существует.

В качестве функции определения порядка обхода вершин F(x) была выбрана функция подсчёта расстояния между двумя точками, заданная следующей формулой:

(1) 
$$D = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

где  $x_1$  <br/>и  $y_1$  — координаты первой точки,  $x_2$  <br/>и  $y_2$  — координаты второй точки

Алгоритм был реализован на языке C++. В данной реализации была добавлена возможность изменения размера ячейки сетки, наложенной на карту. Сетка делит карту на сектора, которые в алгоритме выступают в роли вершин, а размер ячейки соответствует габаритам ПТС. Найденные путь обхода препятствий с разным размером сетки показаны на рис. 4.

#### 6. Алгоритм упрощения пути

Под упрощением пути понимается процедура удаления лишних вершин (точек). Лишней вершиной является та вершина проход через которую не обязателен при движении к конечной вершине. Разработанный алгоритм состоит из следующих этапов:

- (1) выбирается главная вершина. На начальном этапе это самая первая вершина, которая в данном случае является начальной точкой *A*.
- (2) выполняется построение виртуального отрезка между главной вершиной и каждой следующей вершиной в списке.
- (3) если построение виртуального отрезка удалось, то выбирается следующая вершина, а главная остаётся той же, пока виртуальная линия не будет над препятствием.
- (4) если построение виртуального отрезка не удалось (линяя над препятствием), то вершина, которая была перед текущей, выбирается главной. Вершина, которая была главной сохраняется, а все вершины между сохранённой главной и текущей главной считаются лишними. Между сохранённой главной и текущей главной строиться отрезок.
- (5) процесс продолжается пока главной вершиной не станет конечная точка, которая в данном случае является точкой *B*.

После отсеивания лишних вершин, между начерченными отрезками высчитывается значение угла в градусах, который является углом поворота мобильного робота при следовании по найдённому пути, а также направление поворота. Угол высчитывается по следующей формуле:

(2) 
$$\cos(\varphi) = \frac{(v_1, v_2)}{|v_1| * |v_2|} = \frac{x_1 x_2 + y_1 y_2}{\sqrt{x_1^2 + y_1^2} * \sqrt{x_2^2 + y_2^2}}$$

Здесь  $v_1$  и  $v_2$  — векторы образованные отрезками между которыми считается угол. Результат показан упрощения пути показан на рис. 5.



Рис. 5: Слева — найденный путь; справа — упрощенный путь

## 7. Программная реализация

Базой для программной реализации вышеперечисленных алгоритмов стала система распределённой блочно-параллельной обработки данных. Эта система была разработана в ИПС РАН. Основной отличительной чертой системы является использование модульных схем для обработки данных. В основе модульной схемы лежит принцип «разделяй и властвуй», то есть каждый алгоритм заключается в свой модуль. Применительно к рассматриваемой задачи схема является линейным графом последовательно выполняемых модулей:

- (1) модуль **ReadStereo**. Здесь происходит считывание и синхронизация данных, полученный со стереопары камер. На выходе у модуля стереопара изображений.
- (2) модуль **CreateDispMap**. Данный модуль принимает стерео изображение и создаёт карту глубины. Созданная карта глубины, передаётся следующему модулю.
- (3) модуль **CreateBinMap**. Модуль, в котором выполняется детектирование близ лежащих к ПТС объектов, расчёт расстояний до них и нанесение их на бинарную карту сцены.
- (4) модуль **CreatePath**. Здесь к полученной карте сцены применяется алгоритм A\*, а также алгоритм упрощения найденного пути. В данном модуле путь представлен в виде списка опорных точек с указанием угла поворота. Также модуль формирует специальную команду для устройства, которое непосредственно управляет двигателями ПТС.

Следует учесть, что данная схема выполняется циклически, то есть после успешного нахождения пути, ПТС должен совершить необходимые действия для достижения контрольной точки, а затем получить новую стереопару изображений и обновить карту сцены. Однако,



Рис. 6: Базовая конструкция ПТС

прохождение всего пути может вызвать затруднение, так как ПТС двигаясь от начальной до конечной точки не получает новых данных о расположении близлежащих объектах. Поэтому было решено сократить полный путь до двух контрольных точек: стартовая и промежуточная. В промежуточной точке ПТС останавливается и получает новую стереопару изображений.

Система распределённой блочно параллельной обработки данных позволяет использовать алгоритмы предложенного метода на высокопроизводительных кластерных вычислителях. В свою очередь, благодаря модульной реализации данного метода, можно вносить изменения в каждый отдельный модуль, изменяя и/или удаляя его компоненты. Также есть возможность добавить новый модуль обработки в существующую схему в зависимости от решаемой задачи, что делает метод более универсальным по сравнению с аналогами [3,7].

#### 8. Аппаратная реализация

В целях проверки работоспособности алгоритмов была разработана базовая конструкция ПТС. В данном случае ПТС это мобильный робот способный передвигаться во всех четырёх направления. На Рис. 6. показана схема базовой конструкции ПТС.

Базовая конструкция ПТС включает в себя:

- (1) блок сенсоров. В данном блоке расположены различные сенсоры видимого и невидимого диапазона, такие как стереопара камер и дальномеры. Также есть возможность установить более сложные сенсоры, например, лидар.
- (2) бортовой вычислитель. В качестве вычислителя можно использовать одноплатный микрокомпьютер Raspberry Pi или его аналоги, такие как Banana Pi и Orange Pi. Чтобы общаться с любыми внешними устройствами и управлять ими, Raspberry Pi имеет на борту низкоуровневый интерфейс ввода-вывода прямого управления, называемый GPIO (General Purpose Input Output).
- (3) блок двигателей. В данном блоке расположена вся аппаратная начинка, которая позволяет ПТС декодировать команды от бортового вычислителя и совершать движение.
- (4) кластерный вычислитель. При необходимости ПТС может передавать часть данных на обработку в кластерный вычислитель. Таким образом, ресурсоёмкая обработка информации будет занимать меньше времени.

Исходя из схемы базовой конструкции, был создан экспериментальный образец ПТС: рис. 6b.

В качестве основания используется полноприводная платформа из оргстекла с 4 двигателями, позволяющая совершать поворот на месте. Блок сенсоров здесь представлен стереокамерой и цифровым магнитометром, который позволяет отследить угол поворота ПТС. За обработку информации отвечает одноплатный компьютер Raspberry Pi 3, оснащённый Wi-Fi модулем для связи с кластерным вычислителем.

## 9. Практические результаты

Предложенный метод обхода препятствий был протестирован в искусственно созданной среде с использованием реального прототипа ПТС, который был описан ранее. В качестве препятствий использовались картонные коробки. Тестирование проходило в несколько этапов:

(1) запуск схемы на бортовом вычислителе:

- создание карты глубины;
- создание бинарной карты сцены;
- расчёт пути.
- (2) наблюдение за передвижением ПТС в соответствии с полученной командой;
- (3) сравнение рассчитанного пути с реальным.



Рис. 7: Реальный и рассчитанный путь

На первом этапе была получена бинарная карта сцены с нанесённым на неё рассчитанным путём. После того, как команда сформирована и передана на управляющее устройство, ПТС начал своё движение по заданному пути. Далее происходит остановка ПТС, что свидетельствует о его готовности выполнить новую команду. На этом тестирование было завершено. На рис. 7 показан результат сравнения реального и рассчитанного пути.

#### Заключение

Представленный метод обхода препятствий имеет ряд преимуществ по сравнению с методами навигации мобильных роботов по карте глубины, рассмотренных в статьях [8,9]. Среди них, можно отметить следующее:

- открытость технологий и исходного кода;
- использование общедоступного оборудования, такого как обыкновенные Web камеры, без использования специфических RGB-D сенсоров.

Однако, данный метод имеет существенный недостаток, связанный с временем построения карты глубины для единовременно наблюдаемой сцены, которое может достигать 4 секунд, что гораздо дольше, чем у Kinect подобных сенсоров. Тем не менее, данный недостаток может быть частично нивелирован путём установки более производительного бортового вычислителя или путём передачи части вычислений на кластерную установку по средством Wi-Fi соединения. Сравнивая данный метод с методами навигации роботов с использованием УЗ, ИК или лазерных дальномеров [10], можно также выделить преимущества, такие как:

- построение карты сцены не зависит от материала наблюдаемых объектов;
- нет необходимости в системе поворота стереокамеры, так как она имеет больший угол обзора чем дальномер.

Таким образом, был получен метод обхода препятствий на базе стереозрения в задачах навигации мобильных роботов и ПТС. Разработанный метод успешно прошёл испытания с применением реального прототипа ПТС в смоделированной ситуации с использованием системы распределённой блочно-параллельная обработка данных [11]. Однако существенным недостатком данного метода является время построения карты глубины наблюдаемой сцены.

В настоящее время коллективом разработчиков ведётся работа над оптимизацией алгоритмов построения карты глубины, а также проводятся более масштабные испытания предложенного метода для выявления слабых мест.

#### Список литературы

- Т. Кормен, Ч. Лейзерсон, Р. Ривест. «Элементарные алгоритмы для работы с графами», Алгоритмы: построение и анализ, Глава 22, второе издание, «Вильямс», М., 2005. ↑<sup>331</sup>
- G. Ryan, R. Roelofs. Simultaneous Localization and Mapping, Swarthmore College. Dept. of Engineering, 2013, URL: http: //thesis.haverford.edu/dspace/handle/10066/11713<sup>32</sup>
- [3] D. Murray, J. J. Little. "Using Real-Time Stereo Vision for Mobile Robot Navigation", Autonomous Robots, 8:2 (2000), pp. 161–171. <sup>332,340</sup>
- [4] A. Rankin, A. Huertas, L. Matthies. "Evaluation of Stereo Vision Obstacle Detection Algorithms for Off-Road Autonomous Navigation", AUVSI Symp. on Unmanned Systems (Baltimore, MD, June 28–30, 2005), URL: https: //trs.jpl.nasa.gov/bitstream/handle/2014/38362/05-1610.pdf ↑ <sup>332</sup>
- [5] H. Hirschmuller. "Stereo Vision in Structured Environments by Consistent Semi-Global Matching", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. V. 2, CVPR 2006 (New York, NY, USA, June 17–22, 2006), 2006, pp. 2386–2393. ↑ <sup>333</sup>
- [6] Hart P.E., Nilsson N.J., Raphael B.. "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", *IEEE Trans. Systems Science* and Cybernetics, 4:2 (1968), pp. 100–107. ↑ <sup>336</sup>

- [7] F. Duchoň, A. Babineca, M. Kajan, P. Beňo, M. Florek, T. Fico, L. Jurišica.
  "Path Planning with Modified a Star Algorithm for a Mobile Robot", MMaMS 2014 (Vysoke Tatry, Slovakia, November 25–27, 2014), Procedia Engineering, 96 (2014), pp. 59–69. ↑ <sup>340</sup>
- [8] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, R. Siegwart. "Kinect v2 for Mobile Robot Navigation: Evaluation and Modeling", *Proceedings of the 17th International Conference on Advanced Robotics*, ICAR 2015 (Istanbul, Turkey, July 27–31, 2015). ↑ <sup>342</sup>
- [9] A. Schmidt, M. Fularz, M. Kraft, A. Kasiński, M. Nowicki. "An Indoor RGB-D Dataset for the Evaluation of Robot Navigation Algorithms", *Advanced Concepts for Intelligent Vision Systems*, 15th International Conference ACIVS 2013 (Poznań, Poland, October 28–31, 2013), Lecture Notes in Computer Science, vol. 8192, Springer International Publishing, 2013, pp. 321–329. <sup>342</sup>
- [10] Н. С. Сенюшкин, А. В. Суханов, В. В. Парамонов. «Автономная система позиционирования в составе управления наземной многоцелевой подвижной платформой», *Молодой ученый*, 2011, №1, с. 44–46. ↑ <sup>343</sup>
- [11] A. Kondratyev, I. Tishchenko. "Concept of Distributed Processing System of Images Flow", *Robot Intelligence Technology and Applications 4*, Results from the 4th International Conference on Robot Intelligence Technology and Applications (RiTA2015). V. III, Advances in Intelligent Systems and Computing, vol. 447, eds. J.-H. Kim, F. Karray, J. Jo, P. Sincak, H. Myung, Springer International Publishing, 2016, pp. 551–561. ↑ <sup>343</sup>

Рекомендовал к публикации

д.ф.-м.н. С. В. Знаменский

Пример ссылки на эту публикацию:

А. В. Смирнов, А. Ю. Беззубцев. «Обход препятствий подвижными техническими средствами с использованием стереозрения», *Программные системы: теория и приложения*, 2016, **7**:4(31), с. 331–346.

URL: http://psta.psiras.ru/read/psta2016\_4\_331-346.pdf

#### Обход препятствий ПТС с использованием стереозрения

Об авторах:



Александр Владимирович Смирнов

Инженер Лаборатории методов обработки и анализа изображений, Институт Программных Систем имени А. К. Айламазяна PAH

e-mail:

asmirnov 1991@mail.ru



Артём Юрьевич Беззубцев

Инженер Лаборатории методов обработки и анализа изображений, Институт Программных Систем имени А. К. Айламазяна PAH

e-mail:

mannaz2012@mail.ru

Aleksandr Smirnov, Artem Bezzubtsev. Bypass obstacles mobile technical unit using stereo vision.

ABSTRACT. In the article offered the obstacle avoidance method in the way of mobile technical unit (MTU) using stereovision algorithms and distributed data block parallel processing system. The article also describes the algorithm developed card generating the test room, considered the use of A<sup>\*</sup> algorithm to calculate the bypass path, and put forward the concept of creating a real MTU for testing algorithms. (In Russian).

Key words and phrases: Mobile technical unit (MTU), depth map, stereovision, rectification, A\*, distributed system, Raspberry Pi.

#### References

- [1] T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein. Introduction to Algorithms, The MIT Press, 2009.
- [2] G. Ryan, R. Roelofs. Simultaneous Localization and Mapping, Swarthmore College. Dept. of Engineering, 2013, URL: http://thesis.haverford.edu/dspace/handle/ 10066/11713
- [3] D. Murray, J.J. Little. "Using Real-Time Stereo Vision for Mobile Robot Navigation", Autonomous Robots, 8:2 (2000), pp. 161–171.

<sup>©</sup> Program systems: Theory and Applications, 2016

#### А. В. Смирнов, А. Ю. Беззубцев

- [4] A. Rankin, A. Huertas, L. Matthies. "Evaluation of Stereo Vision Obstacle Detection Algorithms for Off-Road Autonomous Navigation", AUVSI Symp. on Unmanned Systems (Baltimore, MD, June 28-30, 2005), URL: https://trs.jpl.nasa.gov/bitstream/handle/2014/38362/05-1610.pdf
- [5] H. Hirschmuller. "Stereo Vision in Structured Environments by Consistent Semi-Global Matching", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. V. 2, CVPR 2006 (New York, NY, USA, June 17–22, 2006), 2006, pp. 2386–2393.
- [6] Hart P.E., Nilsson N.J., Raphael B.. "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", *IEEE Trans. Systems Science and Cybernetics*, 4:2 (1968), pp. 100–107.
- [7] F. Duchoň, A. Babineca, M. Kajan, P. Beňo, M. Florek, T. Fico, L. Jurišica. "Path Planning with Modified a Star Algorithm for a Mobile Robot", MMaMS 2014 (Vysoke Tatry, Slovakia, November 25–27, 2014), *Procedia Engineering*, **96** (2014), pp. 59–69.
- [8] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, R. Siegwart. "Kinect v2 for Mobile Robot Navigation: Evaluation and Modeling", *Proceedings of the 17th International Conference on Advanced Robotics*, ICAR 2015 (Istanbul, Turkey, July 27–31, 2015).
- [9] A. Schmidt, M. Fularz, M. Kraft, A. Kasiński, M. Nowicki. "An Indoor RGB-D Dataset for the Evaluation of Robot Navigation Algorithms", Advanced Concepts for Intelligent Vision Systems, 15th International Conference ACIVS 2013 (Poznań, Poland, October 28–31, 2013), Lecture Notes in Computer Science, vol. 8192, Springer International Publishing, 2013, pp. 321–329.
- [10] N.S. Senyushkin, A.V. Sukhanov, V.V. Paramonov. "Stand-alone positioning system as a part of a control system for multipurpose land vehicle", *Molodoy* uchenyy, 2011, no.1, pp. 44–46.
- [11] A. Kondratyev, I. Tishchenko. "Concept of Distributed Processing System of Images Flow", *Robot Intelligence Technology and Applications 4*, Results from the 4th International Conference on Robot Intelligence Technology and Applications (RiTA2015). V. III, Advances in Intelligent Systems and Computing, vol. 447, eds. J.-H. Kim, F. Karray, J. Jo, P. Sincak, H. Myung, Springer International Publishing, 2016, pp. 551–561.

#### Sample citation of this publication:

Aleksandr Smirnov, Artem Bezzubtsev. "Bypass obstacles mobile technical unit using stereo vision", *Program systems: Theory and applications*, 2016, **7**:4(31), pp. 331–346. (*In Russian*).

URL: http://psta.psiras.ru/read/psta2016\_4\_331-346.pdf