

РЕАЛИЗАЦИЯ И ВНЕДРЕНИЕ АЛГОРИТМА ОБУЧЕНИЯ С ПОДКРЕПЛЕНИЕМ ДЛЯ АВТОНОМНОГО ПЕРЕДВИЖЕНИЯ РОБОТА В ПРОСТРАНСТВЕ

Черникова Василиса Михайловна

Научный руководитель Сухаренко Данил Владимирович

ГАОУ СО «СамЛИТ (Базовая школа РАН)», Самарская область, г. Самара

Сегодня все чаще можно услышать информацию о беспилотных автомобилях и устройствах, которые могут двигаться самостоятельно, при этом, интеллектуально определяющих свое положение и способы передвижения. Данная работа была направлена на реализацию ползающего робота, способного самостоятельно обучиться движению и построению карты перемещений. Для построения конструкции и всех механических частей были спроектированы соответствующие 3D-модели и в дальнейшем распечатаны на 3D-принтере. В качестве электронных компонентов была использована плата Raspberry PI 3 и сервоприводы повышенного усилия (28 кг).

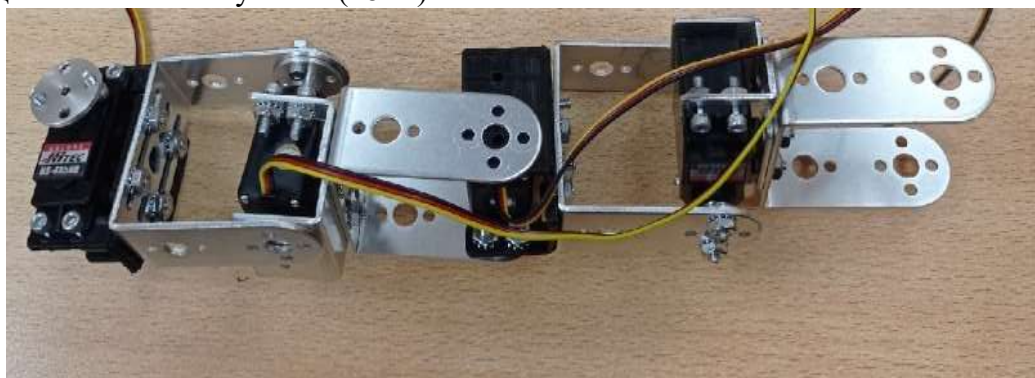


Рис.1. Сборка конструкции

Для осуществления передвижения роботов (шагающих, ползающих, колесных) был рассмотрен алгоритм обучения с подкреплением. Примером такого алгоритма может послужить алгоритм (VPG) - это метод обучения роботов-агентов, чтобы научиться планировать дополнительные толкающие и захватывающие действия для манипуляции (например, для неструктурированных приложений захвата и захвата). VPG работает непосредственно с визуальными наблюдениями (изображения RGB-D), учится методом проб и ошибок, быстро обучается и обобщает для новых объектов и сценариев. Оптимизированный алгоритм был представлен на международной конференции IEEE / RSJ по интеллектуальным роботам и системам (IROS) в 2018 году.

Суть работы алгоритма заключается в следующем, ставится стационарная камера в область зрения, которой помещен сам робот и объекты манипуляции, за программную часть отвечает нейронная сеть, которая позволяет распознать объект и его местоположение, предполагаемые перемещения формируют таблицу (матрицу состояний), чем выгодней для решения задачи положение робота и объекта, тем выше ранг матрицы. В итоге при случайных перемещениях при начальном запуске программы выгодные расположения фиксируются, т.к. они в приоритете, и накапливаются, в случае если робот случайно сдвинулся в невыгодное положение значения отнимаются и в приоритете остается предыдущее максимальное значение выходных значений, робот постепенно обучается и не делает ошибок в перемещении, в случае большого объема итераций.

Для настройки программы сформирован обучающий датасет состоящий из изображений положения робота, который был использован для обучения сверточной нейронной сети для распознавания местонахождения робота.

```

[ ] from tensorflow.keras import layers
    from tensorflow.keras import Model

[ ] # Наша карта входных характеристик имеет размер 150x150x3:
    # 150x150 для пикселей изображения и 3 для трех цветовых каналов: R, G и B
    img_input = layers.Input(shape=(150, 150, 3))

    # Первая свертка извлекает 16 фильтров размером 3x3
    # Свертка сопровождается слоем max-pooling с окном 2x2
    x = layers.Conv2D(16, 3, activation='relu')(img_input)
    x = layers.MaxPooling2D(2)(x)

    # Вторая свертка извлекает 32 фильтра размером 3x3
    # Свертка сопровождается слоем max-pooling с окном 2x2
    x = layers.Conv2D(32, 3, activation='relu')(x)
    x = layers.MaxPooling2D(2)(x)

    # Третья свертка извлекает 64 фильтра размером 3x3
    # Свертка сопровождается слоем max-pooling с окном 2x2
    x = layers.Conv2D(64, 3, activation='relu')(x)
    x = layers.MaxPooling2D(2)(x)

```

Рис.2. Свертка, объявление слоев и их активационных функций

В итоге собрана конструкция робота, способная самостоятельно обучиться движению с течением времени, а также протестирована комбинация программ, состоящих из 2 нейронных сетей и программы базовых перемещений объекта.

Список литературы

1. Learning Synergies between Pushing and Grasping with Self-supervised Deep Reinforcement Learning, Сайт. [URL:\(https://github.com/andyzeng/visual-pushing-grasping\)](https://github.com/andyzeng/visual-pushing-grasping)