

ния, программное обеспечение содержит в себе также функционал для распознавания автомобильных номеров.

Для решения данной задачи есть много подходов. Один из подходов основан на шаблонах. Для этого подхода у автомобильного номера должен быть определенный шрифт, масштаб, также необходимо известное место нахождения номера и жесткий формат. Плюсы этого подхода состоят в том, что достигается очень высокая точность и минимальный процент ошибок. Но у этого подхода есть минус: ограниченность применения. Это выражается в том, что не для любого вида съемки можно применить распознавание, так как не всегда соблюдается шаблон.

Поэтому для более гибкого распознавания используются машинное обучение и нейронные сети. Часто, использование машинного обучения содержит в себе сложности связанные с тренировками, а также нейронные сети менее точны, чем шаблонные методы.

Чаще всего в современном мире используют сверточные нейронные сети (CNN). Благодаря алгоритмам CNN можно делать надежные и гибкие решения для применения. Также при таком подходе за счет переобучения точность и надежность повышаются. Сам алгоритм можно разбить на несколько частей, которые рассчитываются разными нейронными сетями. Первый шаг это поиск номера автомобиля на снимке и определение его границ и центра. Для этого используют несколько CNN. На втором шаге происходит распознавание текста на номере и получение окончательно результата распознавания.

Таким образом, распознавание номеров очень полезная технология для автоматизации многих действий человека. Оно активно используется на различных пунктах пропуска, автостоянках и для контроля правил дорожного движения.

А. Л. Шубов, О. А. Кравченко
(ГГТУ им. П. О. Сухого, Гомель)

ПРИМЕНЕНИЕ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ ПРИ РАЗРАБОТКЕ 2D-ИГР

Требование реализации проекта как *2D-игры* приводит к необходимости анализа применения целого набора современных технологий, которые могут быть использованы для реализации поставленной задачи.

Unity – это межплатформенная среда для разработки компьютерных игр в реальном времени, которая предоставляет вам набор готовых функций при помощи языка *C#*. Появление продвинутых и относительно простых в использовании игровых движков, таких как *Unity*, еще больше упростило создание прототипов, выражающих концепции игрового дизайна [1]. Большим плюсом является наличие визуальной среды разработки и возможность создания игр сразу под несколько платформ.

GameMakerStudio 2 – это первая остановка на пути к разработке игр. Данная платформа проста в освоении и отлично подходит для коротких *2D RPG* и платформеров. В интернете также немало дополнений для движка, позволяющих кастомизировать его для создания платформера или *RPG* с видом сверху.

Unreal Engine представляет собой игровой движок, разрабатываемый и поддерживаемый компанией *Epic Games*. Нередко компании используют этот движок для создания собственного, так как он позволяет работать с хорошей графикой и дает возможность создавать игры под любую платформу. В вопросе создания *2D*-игр выбор данного движка неоднозначен, так как плюсы в виде отличной графики уже не так явно выделяются.

Каждая из сред разработки имеет свои плюсы и минусы в индивидуальном порядке, что предоставляет разработчику широкий выбор среды разработки на его выбор, в зависимости от поставленной задачи.

Литература

1 Бонд, Д. Г. *Unity* и *C#*. Геймдев от идеи до реализации / Д. Г. Бонд. – М.: Бином-Пресс; СПб.: Питер принт, 2019. – 928 с.