

Ставится задача распознавания вида помещения. Применяются аппаратные средства: робот с видеокамерой, свободно передвигающийся по заданным помещениям. Он оснащен видеокамерой, изображение с которой передается в память компьютера. Предварительно вручную создаются семантические сети предметов мебели и других объектов, которые характеризуют назначение комнаты. Кроме того, строятся семантические сети помещения различных категорий (коридор, гостиная, кухня и т.п.).

Подобную задачу ставил создатель теории фреймов М. Минский [1], он начинал с изучения процессов узнавания предмета человеком. Фрейм – это аналог семантической сети. Вначале М. Минский построил фреймы для классификации простых геометрических трехмерных предметов: куба, пирамиды, параллелепипеда. После этого он усложнил задачу, добавив перемещение предметов по комнате и проверял гипотезу: тали эта комната. Кроме того, при фиксации сцены фотокамерой возможно искажение их линзами объектива. Съёмка сцены при движении наблюдателя также усложняла задачу.

Для решения нашей задачи на компьютере установлено приложение, которое сегментирует полученные с камеры изображения. Оно использует библиотеку модулей компьютерного зрения OpenCV, в которой, в частности, представлены модули обработки изображений на языке Python. Так раздел Image Processing этой библиотеки содержит полезные для сегментации модули, такие как Geometric Transformations of Images, CannyEdgeDetection и многие другие [2].

Семантическая сеть предметов мебели и других объектов, характеризующих помещение, строится в виде ориентированного графа, узлы которого представляют объекты, это, например, конструктивные элементы мебели (сиденье, ножка, спинка и пр.) и их атрибуты (расположение относительно других элементов, форма, допустимые размеры и пр.) [3]. Семантическая сеть помещения также является ориентированным графом, описывающим объекты, присущие данной категории комнаты. Все построенные семантические сети хранятся в памяти компьютера.

Компьютерное приложение сегментирует изображение и методом полного перебора проверяет выделенные объекты на их соответствие семантическим сетям предметов. После проверки всех предметов, находящихся в комнате, приложение выполняет полный перебор семантических сетей категорий комнат. Если найдено соответствие сети с набором предметов, то распознавание комнаты заканчивается и приложение переходит к другой комнате.

Возможное практическое использование созданной системы классификации – это ревизия складских помещений с распознаванием предметов и вычислением их количества [4]. Можно фиксировать номера госрегистрации автомобилей, использующих стоянку и записывать моменты их прибытия и убытия. В помещениях, неблагоприятных для человека, возможна как проверка воздуха (температура, влажность, химический состав) так и составление списка предметов в помещении и фиксация изменения их положения. В помещениях с недостатками освещенности можно для распознавания животных использовать инфракрасные видеокамеры.

В развитии предлагаемого алгоритма предполагается изучить возможность автоматического построения семантических сетей для помещений. Для этого будет использоваться нейронная сеть с обучающей выборкой изображений помещений. Автоматическое построение семантических сетей даже таких простых объектов, как предметы мебели, представляется проблематичным ввиду того, что трудно автоматически разделить конструкцию на составляющие ее элементы. Динамические сцены с наличием людей и животных, которые затевают предметы в комнате еще более усложняют задачу.

Требование о решении задачи в реальном времени вызывает необходимость оптимизировать программное обеспечение и применять более мощные вычислительные средства. Кроме того, в этом случае может возникнуть проблема корректировки модулей, заимствованных из библиотеки OpenCV, то есть потребуются создавать собственные модули сегментации сцены.

Список использованных источников:

1. Минский М. Фреймы для представления знаний // М.: Энергия, 1979, С.338.
2. Pronobis A., Martinez M.O., Caputo B., Jensfelt P. Multi-modal semantic place classification // Int. J. Robot. Res. 2010, 29, 298–320.
3. Astua C., Barber R., Crespo J., Jardon A. Object detection techniques applied on mobile robot semantic navigation // Sensors 2014, 14(4), 6734–6757.
4. Ekvall S., Kragic D., Jensfelt P. Object detection and mapping for service robot tasks // Robotica, 2007, 25(2), 175-187.

ОБЗОР ОСНОВНЫХ АРХИТЕКТУР СЕМАНТИЧЕСКОГО СЕГМЕНТИРОВАНИЯ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Куница Е.Ю.

Лукашевич М.М. – к.т.н., доцент

Сегментация, в обработке изображений — это разделение изображения на множество пикселей (суперпикселей). Обычно сегментация используется для выделения объектов и их грани.

При семантической сегментации каждому множеству пикселей ставится в соответствие определённый

класс объектов. Семантическая сегментация является высокоуровневой задачей обработки изображений. Она является даже более сложной, чем задача классификации изображений и поиска объектов, что обусловлено не только необходимостью определения классов объектов, но и выявления их структуры, правильного выделения частей объектов на изображении.

Одним из популярных начальных подходов к глубокому обучению была классификация patch-based, где каждый пиксель отдельно классифицировался на классы с использованием патча изображения вокруг него. Основная причина использования патчей заключалась в том, что в классификационных сетях обычно есть полностью связанные слои и поэтому требуются изображения фиксированного размера.

В 2014 началась популяризация использования FCN нейронных сетей. Так как в данной архитектуре отсутствуют полносвязные слои, это позволило создавать карты сегментации для изображения любого размера. Так же FCN выполняется намного быстрее по сравнению с подходом к классификации патчей. Основная проблема данного подхода является низкое разрешение на выходе.

Однако семантическая сегментация требует точного выравнивания карт классов. Для решения этой проблемы используются архитектуры типа кодер-декодер. Наиболее часто используемой нейронной сетью данной архитектуры является сеть U-Net. Архитектура U-Net сети представляет собой последовательность слоёв свёртка-пулинг, которые сначала уменьшают пространственное разрешение картинки, а потом увеличивают его, предварительно объединив с данными картинки и пропустив через другие слои свёртки. Таким образом, сеть выполняет роль своеобразного фильтра.

Наиболее важными наборами данных для семантической сегментации.

— VOC2012;

— COCO.

Они предоставляют наборы изображений с эталонными масками.

Однако полученные результаты необходимо обработать так, чтобы получить метрики, по которым можно сказать, насколько хорошо выполнена сегментация, определить какая из архитектур подходит для преследуемых целей.

— $\frac{\sum_i n_{ii}}{\sum_i t_i}$ – точность пикселей;

— $\frac{1}{n_{cl}} \sum_i \frac{n_{ii}}{t_i}$ – средняя точность;

— $\frac{1}{n_{cl}} \sum_i \frac{n_{ii}}{t_i + \sum_j n_{ji} - n_{ii}}$ – средняя IoU (Intersection over Union);

— $(\sum_i t_k)^{-1} \sum_i \frac{t_i n_{ii}}{t_i + \sum_j n_{ji} - n_{ii}}$ – частотно-взвешенное IoU.

n_{cl} – количество классов включённые в карту сегментации, n_{ij} – количество пикселей класса i определённых как класс j , t_i – общее количество пикселей класса i .

Наиболее часто используемые фреймворки и библиотеки для разработки приложений машинного обучения, а в частности семантической сегментации являются:

1. Scikit-learn использует Python для математической и научной работы. scikit-learn включает инструменты для многих стандартных задач машинного обучения (таких как кластеризация, классификация, регрессия и т. д.). И так как scikit-learn разрабатывается большим сообществом разработчиков и экспертов по машинному обучению, перспективные новые методы, как правило, включаются в довольно короткий срок.
2. TensorFlow применяется для создания графов потоков данных. Узлы графа представляют собой математические операции, а ребра графа представляют собой многомерные массивы данных (тензоры), которые связаны между ними. Эта гибкая архитектура позволяет развернуть вычисления на один или несколько CPU или GPUs на десктопе, сервере или мобильном устройстве без перезаписи кода. TensorFlow также включает в себя TensorBoard – набор инструментов для визуализации данных. Графы можно создавать при помощи Python и C++.
3. Keras - это высокоуровневый API для нейронных сетей, написанный на Python и способный работать поверх TensorFlow, CNTK или Theano. Он был разработан с упором на возможность быстрого экспериментирования. Keras позволяет легко и быстро создавать прототипы (благодаря удобству, модульности и расширяемости). Поддерживает как сверточные сети, так и повторяющиеся сети, а также комбинации этих двух.

Список использованных источников:

1. Jonathan Long, Evan Shelhamer, Trevor Darrell, Fully Convolutional Networks for Semantic Segmentation, 2015.
2. Fei-Fei Li, Andrej Karpathy, Justin Johnson, Segmentation and Attention, Lecture 13, 2016.
3. Sasank Chilamkurthy, Guide to Semantic Segmentation with Deep Learning, 2017.