ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

***«*САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО»**

Институт компьютерных наук и технологий

**Высшая школа программной инженерии**

Изображение выглядит как текст

Автоматически созданное описание

**ПРАКТИЧЕСКАЯ РАБОТА №8**

по дисциплине «Проектирование интеллектуальных систем управления»

Студент А. М. Потапова

гр. 3530202/90202

Преподаватель Bahrami AmirHossein

Санкт-Петербург

2022 г

**Введение**

Необходимо создать сверточную нейронную сеть, которая умеет распознавать количество пальцев рук(от 0 до 2). Так же к этой сети необходимо самостоятельно создать тренировочный датасет.

**Ход работы**

Подготовим датасет для обучения нейронной сети:

c = webcam;  
x = 0;  
y = 0;  
height = 400;  
width = 400;  
bboxes = [x y height width]; temp = 0;

while temp <=400

e = c.snapshot;

IFaces = insertObjectAnnotation(e, "rectangle", bboxes, 'Processing Area'); imshow(IFaces);  
filename=strcat(num2str(temp), '.bmp');  
es=imcrop(e, bboxes);

es = imresize(es, [227 227]); imwrite(es, filename);  
temp = temp + 1;  
drawnow;

end

clear c;

Создадим по 400 фотографий для каждого класса(0 пальцев, 1 палец, 2 пальца)

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как текст

Автоматически созданное описание

**Обучение нейронной сети:**

allImages = imageDatastore("Database", "IncludeSubfolders",true, "LabelSource","foldernames");

Выбираем все картинки, которые мы подготовили.

**Структура нейронной сети:**

layers = [  
imageInputLayer([227 227 3]) convolution2dLayer(5,8,'Padding','same') batchNormalizationLayer  
reluLayer

maxPooling2dLayer(2,'Stride',2)

convolution2dLayer(6,20,'Padding','same') batchNormalizationLayer  
reluLayer

maxPooling2dLayer(2,'Stride',2)

fullyConnectedLayer(3) softmaxLayer classificationLayer

];

Описание каждого слоя:

* Image Input Layer.

Входной слой изображения - это место, где вы указываете размер изображения, который в данном случае составляет 227 на 227 на 3. Эти числа соответствуют высоте, ширине и размеру канала. Для цветного изображения размер канала равен 3, что соответствует значениям RGB.

* Convolutional Layer.

В сверточном слое первым аргументом является размер фильтра, который представляет собой высоту и ширину фильтров, используемых обучающей функцией при сканировании изображений. Вы можете указать различные размеры для высоты и ширины фильтра. Второй аргумент - это количество фильтров, numFilters, которое представляет собой количество нейронов, подключающихся к одной и той же области ввода. Этот параметр определяет количество карт объектов. Используйте пару "Заполнение" имя-значение, чтобы добавить заполнение к карте входных объектов. Для сверточного слоя с шагом по умолчанию, равным 1, "одинаковое" заполнение гарантирует, что размер пространственного вывода совпадает с размером ввода. Вы также можете определить шаг и скорость обучения для этого слоя, используя аргументы пары имя-значение convolution2dLayer.

* Batch Normalization Layer.

Слои пакетной нормализации нормализуют активации и градиенты, распространяющиеся по сети, что упрощает задачу оптимизации сетевого обучения. Используйте слои пакетной нормализации между сверточными слоями и нелинейностями, такими как слои ReLU, чтобы ускорить обучение сети и снизить чувствительность к инициализации сети. Используйте слой пакетной нормализации для создания слоя пакетной нормализации.

* ReLU Layer.

За уровнем пакетной нормализации следует нелинейная функция активации. Наиболее распространенной функцией активации является выпрямленный линейный блок (ReLU). Используйте realplayer для создания слоя ReLU.

* Max Pooling Layer.

За сверточными слоями (с функциями активации) иногда следует операция понижающей дискретизации, которая уменьшает пространственный размер карты объектов и удаляет избыточную пространственную информацию. Понижающая выборка позволяет увеличить количество фильтров в более глубоких сверточных слоях без увеличения требуемого объема вычислений на слой. Одним из способов уменьшения выборки является использование максимального пула, который вы создаете с помощью maxPooling2dLayer. Уровень максимального объединения возвращает максимальные значения прямоугольных областей входных данных, указанные первым аргументом poolSize. В этом примере размер прямоугольной области равен [2,2]. Аргумент пары имя- значение 'Stride' задает размер шага, который принимает обучающая функция при сканировании входных данных.

* Fully Connected Layer.

За сверточными слоями и слоями с понижающей выборкой следуют один или несколько полностью связанных слоев. Как следует из названия, полностью связанный слой - это слой, в котором нейроны соединяются со всеми нейронами предыдущего слоя. Этот слой объединяет все особенности, изученные предыдущими слоями по всему изображению, чтобы идентифицировать более крупные узоры. Последний полностью связанный слой объединяет функции для классификации изображений. Следовательно, параметр выходного размера в последнем полностью подключенном слое равен количеству классов в целевых данных. В этом примере размер вывода равен 10, что соответствует 10 классам. Используйте fullyConnectedLayer для создания полностью связанного слоя.

* Softmax Layer.

Функция активации softmax нормализует выходные данные полностью подключенного слоя. Выходные данные слоя softmax состоят из положительных чисел, сумма которых равна единице, которые затем могут быть использованы классификационным слоем в качестве вероятностей классификации. Создайте слой softmax с помощью функции softmax Layer после последнего полностью подключенного слоя.

* Classification Layer.

Последний слой — это слой классификации. Этот уровень использует вероятности, возвращаемые функцией активации softmax для каждого входа, чтобы назначить вход одному из взаимоисключающих классов и вычислить потери. Чтобы создать слой классификации, используйте classificationLayer.

Установка параметров обучения сети.

После определения структуры сети мы указали параметры обучения. Обучили сеть, используя стохастический градиентный спуск с импульсом (SGDM) с начальной скоростью обучения 0,0001. Установили максимальное количество эпох равным 15. Эпоха - это полный цикл обучения для всего обучающего набора данных. Контролируйте точность сети во время обучения, задавая данные проверки и частоту проверки. Перемешивали данные каждую эпоху. Программное обеспечение обучает сеть на основе обучающих данных и вычисляет точность данных проверки через регулярные промежутки времени во время обучения. Данные проверки не используются для обновления весовых коэффициентов сети. Включили график хода обучения и отключите вывод командного окна.

options = trainingOptions("sgdm", ...

"LearnRateSchedule","piecewise", ...

"LearnRateDropFactor",0.2, ...

"LearnRateDropPeriod",5, ...

"MaxEpochs",15, ...

"MiniBatchSize", 100, ...

"InitialLearnRate",0.0001, ...

"Plots","training-progress");

Обучаем нейронную сеть и сохраняем обученную нейронную сеть в файл.

convnet = trainNetwork(allImages, layers, options);

save convnet;

Изображение выглядит как стол

Автоматически созданное описание

Тестируем нашу нейронную сеть:

load convnet;  
x = 0;  
y = 0;  
height = 200;  
width = 200;  
bboxes=[x y height width]; c = webcam;

while true

e = c.snapshot;

IFaces = insertObjectAnnotation(e, "rectangle", bboxes, 'Processing Area'); imshow(IFaces);  
es=imcrop(e, bboxes);  
es = imresize(es, [227 227]);

label = classify(convnet, es); imshow(IFaces); title(char(label));  
drawnow;

end

Изображение выглядит как текст, внутренний

Автоматически созданное описание

Изображение выглядит как текст, внутренний

Автоматически созданное описание

Изображение выглядит как текст, стена, внутренний

Автоматически созданное описание

**Вывод**

В ходе выполнения этого задания мы подготовили датасет для сверточной нейронной сети, которая умеет распознавать, какое количество пальцев показано на фотографии. Так же была написана и обучена такая нейронная сеть. Кроме того, были проведены тесты, подтверждающие ее работоспособность.