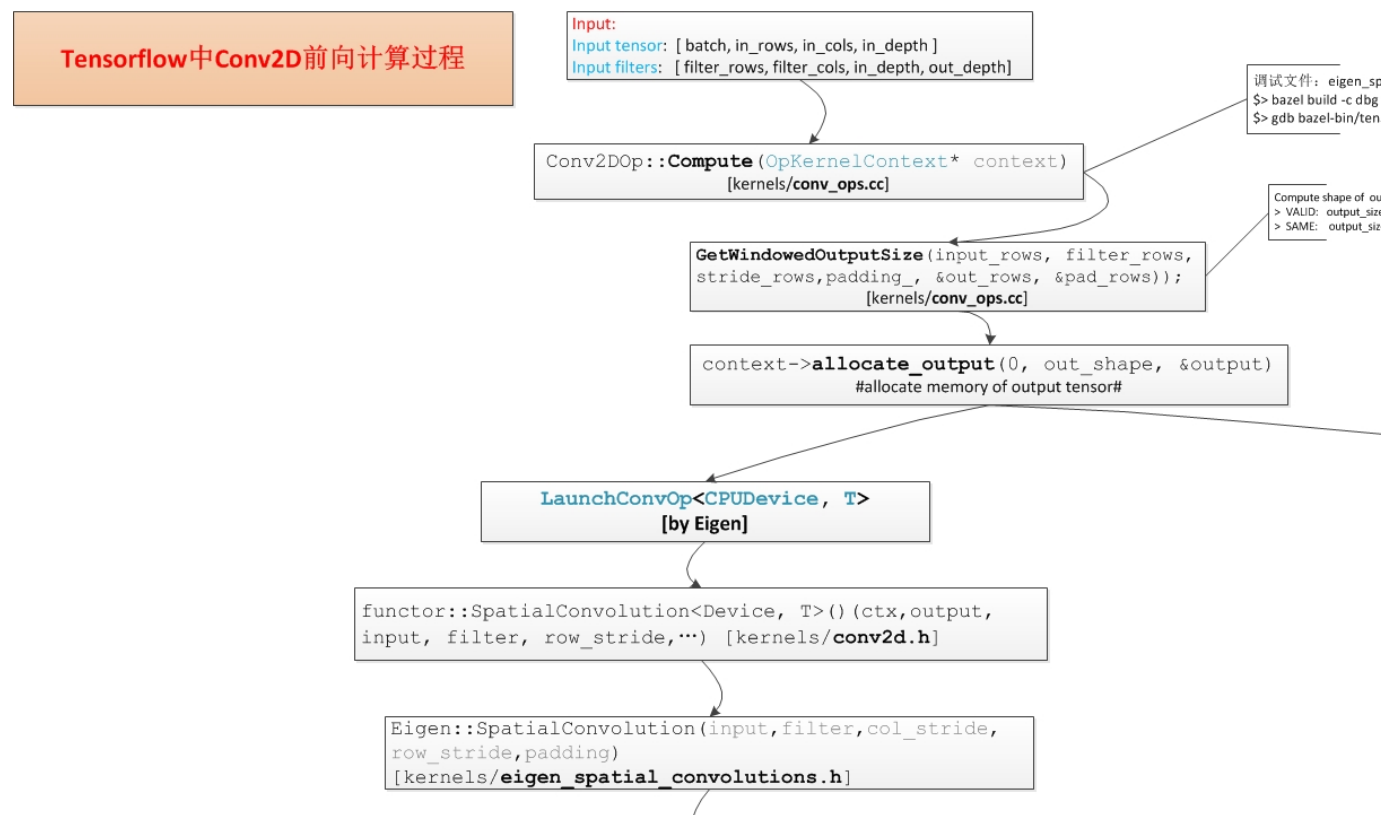


yj0306

博客园 首页 新随笔 联系 订阅 管理

随笔 - 15 文章 - 0 评论 - 2

[tensorflow源码分析] Conv2d卷积运算（前向计算，反向梯度计算）



公告

昵称：yj0306
 园龄：7个月
 粉丝：14
 关注：0
 +加关注

2017年3月						
<	日	一	二	三	四	五
	26	27	28	1	2	3
	5	6	7	8	9	10
	12	13	14	15	16	17
	19	20	21	22	23	24
	26	27	28	29	30	31
	2	3	4	5	6	7

搜索

```

choose(
  Cond<internal::traits<Input>::Layout == ColMajor>(),
  kernel.reshape(kernel_dims)
    .contract(input
      .extract_image_patches(
        kernelRows, kernelCols, row_stride, col_stride,
        row_in_stride, col_in_stride, padding_type)
      .reshape(pre_contract_dims),
        contract_dims)
    .reshape(post_contract_dims),
  input
    .extract_image_patches(kernelRows, kernelCols, row_stride, col_stride,
      row_in_stride, col_in_stride, padding_type)
    .reshape(pre_contract_dims)
    .contract(kernel.reshape(kernel_dims), contract_dims)
    .reshape(post_contract_dims)); [kernels/eigen_spatial_convolutions.h]

```

Conv2d - Eigen 中涉及的变量说明:

```

> Input_tensor: [batch, in_rows, in_cols, in_depth]
> Input_filters: [filter_rows, filter_cols, in_depth, out_depth]
> transform: (in case: row-major order)
  >> out_height, out_width根据padding类型求解
    >>> VALID: output_size = (input_size - filter_size + stride) / stride;
    >>> SAME: output_size = (input_size + stride - 1) / stride;
  >> Input_tensor: [batch * out_height * out_width, in_depth * filter_rows * filter_cols]
  >> Input_filters: [filter_rows * filter_cols * in_depth, out_depth]
  >> output_tensor = input_tensor.contract(input_filters)
最后output_tensor = [batch * out_height * out_width, out_depth]
= [batch, out_height, out_width, out_depth]

```

Tensorflow中Conv2D梯度函数定义

```

Status Conv2DGrad(const AttrSlice& attrs, FunctionDef* g) {
  // clang-format off
  *g = FDH::Define(
    // Arg defs
    {"input: T", "filter: T", "grad: T"},
    // Ret val defs
    {"input_grad: T", "filter_grad: T"},
    // Attr defs
    {"T: {float, double}",
     "strides: list(int)",
     "use_cudnn_on_gpu: bool = true",
     GetPaddingAttrString(),
     GetConvnetDataFormatAttrString()},
    // Nodes
    {
      {"i_shape", "Shape", {"input"}, {"T", "$T"}},
      {"input_grad", "Conv2DBackpropInput", {"i_shape", "filter", "grad"},

```

```

1. Conv2DGrad梯度函数定义中依赖Conv2DBackpropInputOp和Conv2DBackpropInputOp
2. Conv2DBackpropInputOp和Conv2DBackpropInputOp
  > slow-GPU版: class Conv2DSlow
  > CPU版又有两种实现形式
    >> custom: class Conv2DCustom
    >> fast: class Conv2DFastBackprop
  > Custom-CPU版基于Yangqing
  > Fast-CPU版基于Eigen计算库,
  > Slow-GPU版支持GPU下编译运行
3. 设output=Conv2d(input, filter), 则
  > grad(input) = grad(output)
  > grad(filter) = grad(output)

```

```

REGISTER_KERNEL_
  .DevI
  .Host
  Conv2

```

常用链接

我的随笔
我的评论
我的参与
最新评论
我的标签

我的标签

源码 (13) 图解tensorflow (9)
tensorflow (4) graph (2)
session (2) simple placer (1)
线程池模块 (1)
graph constructor (1)
graph optimizer (1) Load (1)
更多

随笔分类

tensorflow源码分析(14)

随笔档案

2016年8月 (15)

相册

tech

最新评论

1. Re:[图解tensorflow源码] Si...

您好，多谢您的分享。上面那个图好像是一个PDF里面的，您有原始PDF么??

--qingyuanxingsi

```

/*Attrs=*/{{"T", "$T"},
           {"strides", "$strides"},
           {"padding", "$padding"},
           {"data_format", "$data_format"},
           {"use_cudnn_on_gpu", "$use_cudnn_on_gpu"}},

{{"f_shape", "Shape", {"filter"}, {""T", "$T"}},
{"filter_grad", "Conv2DBackpropFilter", {"input", "f_shape", "grad"}},
/*Attrs=*/{{"T", "$T"},
           {"strides", "$strides"},
           {"padding", "$padding"},
           {"data_format", "$data_format"},
           {"use_cudnn_on_gpu", "$use_cudnn_on_gpu"}},

});
// clang-format on
return Status::OK();
}
REGISTER_OP_GRADIENT("Conv2D", Conv2DGrad);
[ops/nn_grad.cc]

```

```

REGISTER_KERNEL_BUILDER(Name("Conv2DBackpropFilter")
    .Device(DEVICE_GPU).TypeConstraint<Eigen::half>("T")
    .HostMemory("filter_sizes"),
    Conv2DSlowBackpropFilterOp<GPUDevice, Eigen::half>);
[kernels/conv_grad_ops.cc]

```

类似于Conv2dSlowBackpropFilterOp的计算过程，略了^_^

```

Conv2DBackpropComput
filter_shape, on

```

```

Stream::ThenCon
output_desc,
&scratch_all

```

```

CudnnSupport:
    stream, C
    output_de
    input_des
    algorithm

```

```

dynload::cudnn
dynload::cudnn
dynload::cudnn

```

2. Re:[图解tensorflow源码] 入门...
学习。

--iByte

阅读排行榜

1. [图解tensorflow源码] [原创...]
2. [图解tensorflow源码] 入门...
3. [图解tensorflow源码] TF系...
4. [图解tensorflow源码] Sessi...
5. [图解tensorflow源码] Grap...

评论排行榜

1. [图解tensorflow源码] Simp...
2. [图解tensorflow源码] 入门...

推荐排行榜

1. [图解tensorflow源码] 入门...
2. [图解tensorflow源码] [原创...]

github: <https://github.com/yao62995>

email: yao_62995@163.com

working @ 360_skyeye_lab.

分类: tensorflow源码分析

标签: tensorflow, 源码, conv2d

好文要顶

关注我

收藏该文



yj0306

关注 - 0

粉丝 - 14

0

0

[+加关注](#)[« 上一篇：tensorflow节点布放（device assignment of node）算法：simpler_placer](#)[» 下一篇：\[图解tensorflow源码\] MatMul 矩阵乘积运算（前向计算，反向梯度计算）](#)

posted @ 2016-08-15 15:09 yj0306 阅读(955) 评论(0) 编辑 收藏

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问网站首页](#)。

【推荐】50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库

【招募】2017云栖大会深圳峰会火热报名中

【推荐】阿里云云盾帮您解读WAF防护功能



最新IT新闻:

- 马斯克：特斯拉将从4月开始接受太阳能屋顶预订服务
 - 前商务部长陈德铭爆料：马云丈母娘的钱也存在支付宝里
 - 百度总裁、搜狗CEO都说人工智能目前不可能取代人类
 - 继与去哪儿合作后，斗鱼再联合携程推旅游直播节目
 - 微医向浙江大学捐赠1亿元人民币，主攻医学人工智能
- » 更多新闻...



最新知识库文章:

- 程序员学习能力提升三要素
- 为什么我要写自己的框架？
- 垃圾回收原来是这么回事
- 「代码家」的学习过程和学习经验分享
- 写给未来的程序媛
- » 更多知识库文章...

Copyright ©2017 yj0306