|  |  |
| --- | --- |
| FRCPP前期准备 | 文件类型：技术文档 |
| 文件编号： |
| 面向的部门： 研发中心 |
| 保密等级：高 |
| 作者： 刘凯 |
| 日期：2019-03-6 |
| 版本：1.0V |

FRCPP前期准备



目 录

[1 前期准备 1](#_Toc1400878)

[2 开发框架 1](#_Toc1400879)

1. 深度学习前向架构（讨论）

网络结构

解析

网络初始化

Data

Relu

Loss

Fc

Conv

Output Blob

…...

Input Blob

Output Blob

WeightBlob

Input Blob

Output Blob

WeightBlob

Input Blob

Output Blob

WeightBlob

Input Blob

Layers

Blobs

Net

输入图片

解析

预处理

权重文件

解析

初始化填充

2. 主要的结构和类(讨论)

//定义在Layer.hpp，为层参数

struct LayerParam{

int conv\_stride;

int conv\_pad;

int conv\_width;

int conv\_height;

int conv\_kernels;

int pool\_stride;

int pool\_width;

int pool\_height;

int fc\_kernels;

}

//定义在Net.hpp，为网络参数

struct NetParam{

int batch\_size;

vector <string> layers\_name;

vector <string> layers\_type;

unordered\_map<string, Param> lparams;

void readNetParam(string file);

}

//Layer.hpp，定义基类Layer，后面的conv、relu、pool等均继承Layer。

class Layer{

public:

//构造函数

Layer(){}

//析构函数

virtual ~Layer(){}

//初始化层，包括初始化层参数、填充权重、输入输出维度等

virtual void initLayer() = 0;

//计算输出维度

virtual void calcShape() = 0;

//前向计算

virtual void forward() = 0;

private:

//层参数

LayerParam layer\_param\_;

//权重参数

vector<shared\_ptr<Blob>> weight\_blob\_;

//输入维度

vector<int> inShape\_;

//输出维度

vector<int> outShape\_;

//是否加偏置

bool isBias\_;

}

class ConvLayer : public Layer

class ReluLayer : public Layer

class AffineLayer : public Layer

……

//Blob容器，实现基本的存储、索引、运算等操作

class Blob{

public:

//1.subBlob方法

//2.Blob切片

//3.重载Blob运算符，如+、\*、sum等

//4.返回Blob的维度

private:

int N\_;

int C\_;

int H\_;

int W\_;

vector<cube> blob\_data\_;

}

class Net{

public:

//1.初始化网络，读取参数，填充weight，为特征分配内存

//2.Forward（）控制前向计算

private:

vector<shared\_ptr<Layer>> layers\_;

vector<string> layer\_names\_;

map<string, int> layer\_names\_index\_;

vector<shared\_ptr<Blob > > blobs\_;

vector<string> blob\_names\_;

map<string, int> blob\_names\_index\_;

vector<vector<Blob\*> > bottom\_vecs\_;

vector<vector<int> > bottom\_id\_vecs\_;

vector<vector<Blob\*> > top\_vecs\_;

vector<vector<int> > top\_id\_vecs\_;

}

int main（int argc, char\*\* argv）

{

//1.读取网络和参数文件

//2.读取图片，预处理

//3.定义网络，并初始化

Net myNet(string netFile, string weightFile);

//4.网络前向传播

myNet.Forward(string imgFile);

//5.处理输出结果

}

3. 其他事项

1）名称空间：zq

2）矩阵计算库：Armadillo or Eigen

3）网络、权重和图片的读取和解析

4）代码规范（命名规则、注释等）