# 使用OpenCV可视化caffemodel

## 所须文件

## 代码

#include <opencv2/dnn.hpp>

#include <opencv2/imgproc.hpp>

#include <opencv2/highgui.hpp>

using namespace cv;

using namespace cv::dnn;

#include <fstream>

#include <iostream>

#include <cstdlib>

using namespace std;

int main(int argc, char \*\*argv)

{

String modelTxt="mnist\_deploy.prototxt";

String modelBin="lenet\_iter\_10000.caffemodel";

Ptr<dnn::Importer> importer;

try

{

importer=dnn::createCaffeImporter(modelTxt,modelBin);

}

catch(const cv::Exception &err)

{

std::cerr<<err.msg<<std::endl;

}

if(!importer)

{

std::cerr<<"cant load network!"<<std::endl;

return 0;

}

dnn::Net net;

importer->populateNet(net);

importer.release();

Mat img = imread("4.bmp",0);

if(img.empty())

{

std::cerr<<"cant load image!"<<std::endl;

return 0;

}

resize(img,img,Size(28,28));

dnn::Blob inputBlob = dnn::Blob(img);

net.setBlob(".data",inputBlob);

net.forward();

dnn::Blob prob=net.getBlob("conv1");

for(int i=0;i<20;i++)

{

Mat featureImg=prob.getPlane(0,i);

featureImg.convertTo(featureImg,CV\_8UC1);

resize(featureImg,featureImg,Size(256,256));

imshow("feature",featureImg);

waitKey(0);

}

}

## 编译

g++ -o featureMap featureMap.cpp -lopencv\_dnn -lopencv\_highgui -lopencv\_imgcodecs -lopencv\_imgproc -lstdc++ -lopencv\_core -L/usr/local/lib



## 运行

./featureMap



 