

基于 Lire 的分析与扩展 扩展与展示计划进度

Version 1.0

小组成员:

刘少凡

宋昱材

吴沂楠

黄飞

版本变更记录

版本	变更时间	修改人	审核人	备注
1.0	20170516	刘少凡		从原 1.3 文档中分离出计划进度相关内容单独写在此文档

目录

- 1 具体分工.....4
 - 1.1 第 9 周开发任务分配.....4
 - 1.2 第 10 周开发任务分配.....4
- 2 实现进度.....5
 - 2.1 第 9 周实现进度.....5
 - 2.2 第 10 周实现进度.....9

1 具体分工

1.1 第 9 周开发任务分配

(1) Caffe 环境配置

负责人：宋昱材

工作内容：在 Ubuntu 14.04 操作系统上完成对 Caffe 框架的搭建。

(2) Lire 代码修改

负责人：刘少凡、黄飞

工作内容：完成继承 LireFeature 接口的 CNN 类的具体方法编写。

(3) 界面编写

负责人：吴沂楠

工作内容：完成主界面、入库、检索界面的编写

1.2 第 10 周开发任务分配

(1) CNN 模型调用代码编写

负责人：宋昱材

工作内容：完成调用 CNN 模型的 C++代码编写，实现 JNI 机制。与其他组员配合实现系统整合和调试。

(2) Lire 代码修改与系统整合

负责人：刘少凡、黄飞

工作内容：完成 GetCNN 类的相关代码编写，完成入库和检索模块相关代码的修改。与其他组员配合实现系统整合和调试。

(2) 界面编写

负责人：吴沂楠

工作内容：完成检索结果展示界面的编写。与其他组员配合实现系统整合和调试。

2 实现进度

2.1 第 9 周实现进度

(1) Caffe 环境配置

负责人：宋昱材

● 实现内容

➤ 完成 Caffe 框架的搭建

在 Ubuntu 14.04 上进行了 Caffe 框架的搭建，并进行了调用模型的测试，如图 5.1 所示，为 Caffe 加载 VGG-F 模型后的日志记录。

```
WARNING: Logging before InitGoogleLogging() is written to STDERR
I0504 19:42:14.783478 2470 upgrade_proto.cpp:51] Attempting to upgrade input file specified using deprecated V1LayerParameter: VGG_CNN_F_deploy.prototxt
I0504 19:42:14.783663 2470 upgrade_proto.cpp:59] Successfully upgraded file specified using deprecated V1LayerParameter
I0504 19:42:14.783886 2470 net.cpp:49] Initializing net from parameters:
name: "VGG_CNN_F"
input: "data"
input_dim: 1
input_dim: 3
input_dim: 224
input_dim: 224
state {
  phase: TEST
}
layer {
  name: "conv1"
  type: "Convolution"
```

➤ 确定 CNN 模型

另一个深度学习框架 MatConvNet 总结了近几年的多种 CNNs 模型在相同硬件条件下处理 ILSVRC 2012 测试数据集的准确率与时间效率，其中部分数据如表 5.1 所示。

表 5.1 中的 alexnet 模型是最早提出的 CNNs 模型。vgg-f 模型沿用了 alexnet 模型的 8 层基础架构，并通过调整卷积核的尺寸和数量，改变参数数量，使得模型在保持并一定程度上提升分类准确率的情况下，很大程度提高了处理图片的时间效率。

模型在实际运行过程中处理图片的时间复杂度直接影响实际应用系统运行时特征提取的时间效率，因此模型的时间复杂度不能过高。

基于以上原因，确定了基于 8 层基本 CNNs 结构的、时间复杂度较低的 VGG-F 模型作为项目所选择的模型。

表 5.0-1 各 CNNs 模型准确率与时间效率对比

模型名称	提出时间	模型层数	Top-1 错误率	Top-5 错误率	每秒处理图片量
resnet-50 ^[26]	2015	50	24.6	7.7	315.3
resnet-101 ^[26]	2015	101	23.4	7.0	212.7
resnet-152 ^[26]	2015	152	23.0	6.7	156.6
vgg-verydeep-19 ^[14]	2014	19	28.7	9.9	154.5
vgg-verydeep-16 ^[14]	2014	16	28.5	9.9	183.1
googlenet ^[27]	2014	22	34.2	12.9	501.8
vgg-f ^[28]	2013	8	41.1	18.8	793.1
alexnet ^[13]	2012	8	42.6	19.6	382.4

注：表中数据来源为 <http://www.vlfeat.org/matconvnet/pretrained/>。

● 下周工作

完成调用 CNN 模型的 C++代码编写，实现 JNI 机制。与其他组员配合实现系统整合和调试。

(2) Lire 代码修改

负责人：刘少凡、黄飞

● 实现内容

通过继承 LireFeature 接口，实现了 CNN 类，主要工作为实现如图所示的相关方法。

```
public class CNN implements LireFeature{
    double[] d_feature_long = new double[4096];
    double[] d_feature_long_new = new double[4096];
    double[] d_feature_short = new double[128];
    float[] f_feature_long = new float[4096];

    byte[] feature_byte = new byte[1024];

    GetCNN myGetCNN = new GetCNN();
    DimensionReduction myReduction = new DimensionReduction();

    public void extract(BufferedImage image)
    {
    public String getFeatureName()
    {

    public String getFieldNames() {
    public double[] getDoubleHistogram() {
    public byte[] getByteArrayRepresentation()
    {
    public void setByteArrayRepresentation(byte[] featureData)
    {
    public void setByteArrayRepresentation(byte[] in, int offset, int length)
    {

    public float getDistance(LireFeature vd)
    {

    public String getStringRepresentation() { // added by mlux
    public void setStringRepresentation(String s) { // added by mlux

    private double float2double(float f)
    {
```

- 下周工作

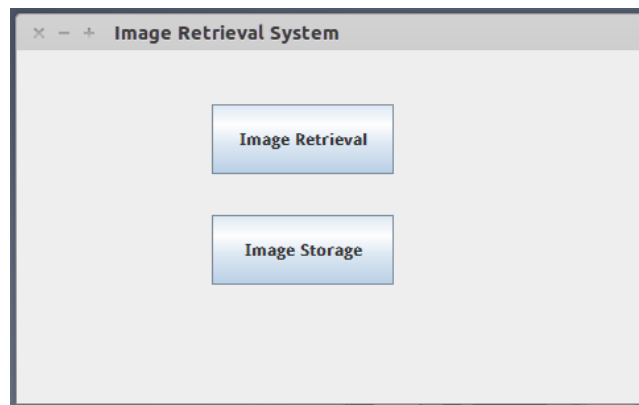
完成 GetCNN 类的相关代码编写，完成入库和检索模块相关代码的修改。与其他组员配合实现系统整合和调试。

- (3) 界面编写

负责人：吴沂楠

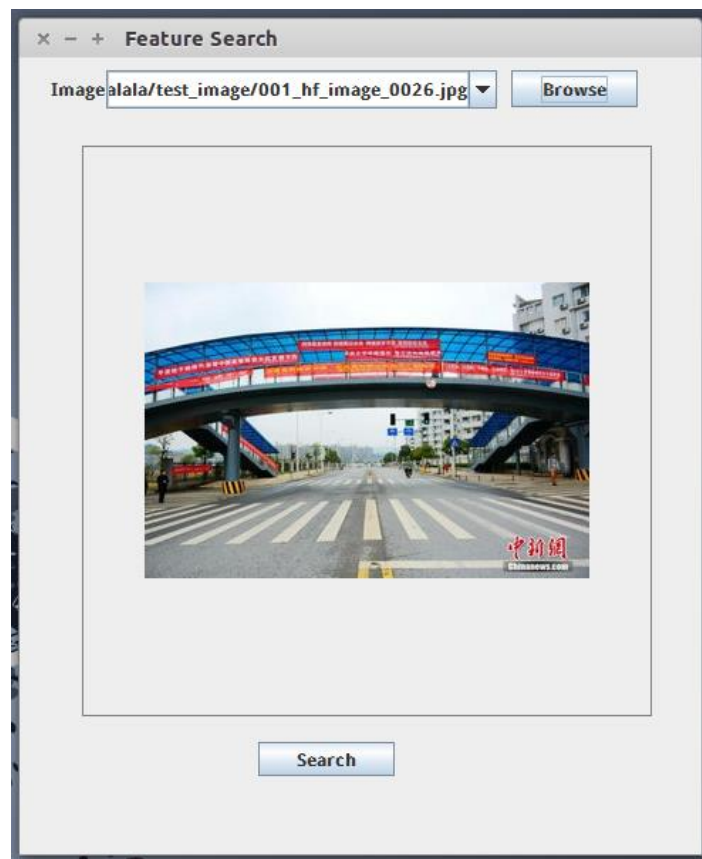
- 实现内容

- 主界面



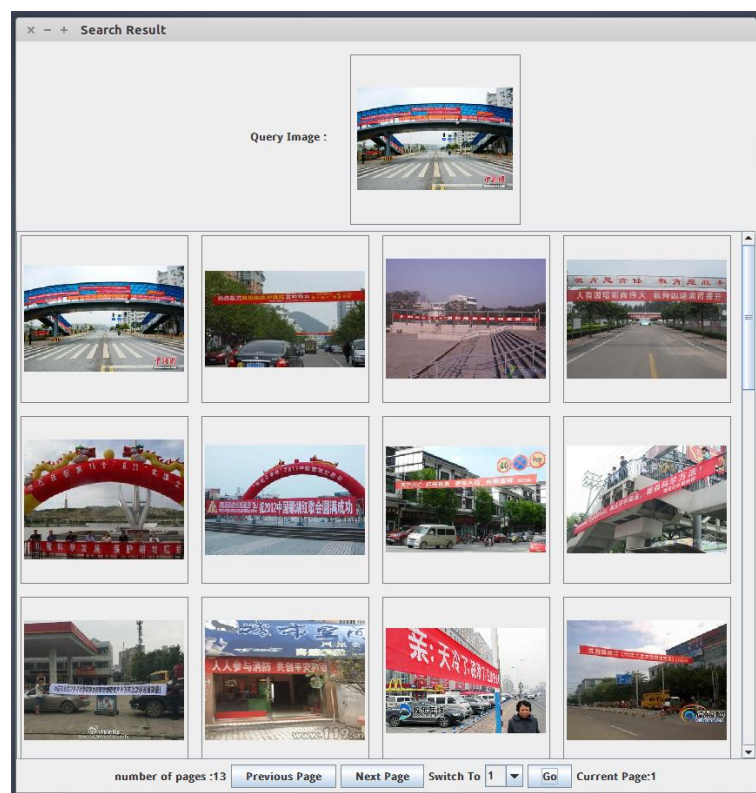
通过点击检索和入库按钮，分别进入检索界面和入库界面

- 检索界面



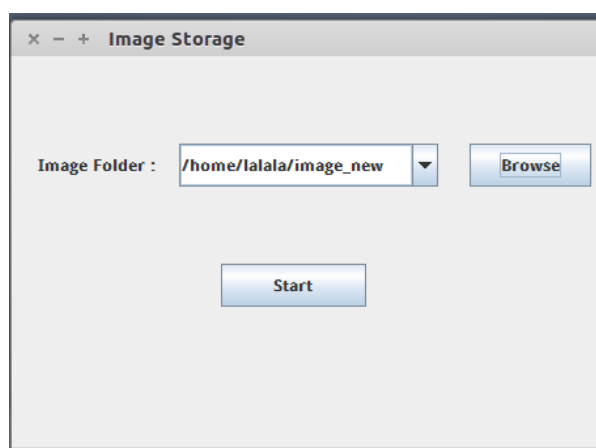
点击 **Browse** 按钮，弹出文件浏览窗口，用户点击文件夹进入存放检索图片的位置，文件浏览窗口自动过滤只保留 jpg 格式图片并显示在文件浏览窗口中，用户从中选择检索示例图片，点击检索按钮，检索结果由结果展示结果显示。

➤ 检索结果展示界面



界面上方显示用户输入的实例图片，下方显示检索结果。

➤ 入库界面



用户输入入库图片文件夹的路径，点击开始，进行入库。

● 下周工作

协助宋昱材完成调用 CNN 模型的 C++代码编写，实现 JNI 机制。与其他组

员配合实现系统整合和调试。

2.2 第 10 周实现进度

(1) GetCNN 类实现

负责人：刘少凡、黄飞

实现 GetCNN 类, GetCNN 类主要实现两个 native 方法, loadCNN 和 getCNN 方法。相关代码如图所示。

```
package net.semanticmetadata.lire.imageanalysis.cnn;
import java.io.File;
import java.io.IOException;
import javax.imageio.ImageIO;
import java.awt.image.BufferedImage;
public class GetCNN {
    String NetTxt = "/home/lalala/TrainedModel/vgg/VGG_F/VGG_CNN_F_deploy.prototxt";
    String NetPara = "/home/lalala/TrainedModel/vgg/VGG_F/VGG_CNN_F_caffemodel";
    String layerName = "fc7";

    int i = 0;

    public GetCNN()
    {
        loadCNN(NetTxt, NetPara);
    }

    public native void loadCNN(String NetTxt, String NetPara);
    public native float[] getCNN(String ImagePath, String layerName);
    static{
        System.loadLibrary("Hello");
    }
    public float[] getFeature(BufferedImage image)
    {

```

loadCNN 方法输入模型定义文件路径和模型参数文件路径,通过 C++代码,将 CNN 模型加载到内存中。

getCNN 方法输入图片路径和层数,通过 C++代码,调用内存中的 CNN 模型提取图像特征。

getFeature 方法为提供给 CNN 类的外部接口, CNN 类通过 getFeature 方法调用 getCNN 方法提取特征。

调用 javah 命令,生成 GetCNN 类对应的 C++头文件。

修改图像入库和图像检索的相关代码,增加 CNN 特征相关的接口。

(2) C++代码实现

负责人：宋昱材、吴沂楠

通过包含 GetCNN 类生成的头文件,实现 C++的函数,主要为两个函数,即 GetCNN 类中 loadCNN 方法和 getCNN 方法对应的 C++函数。

Java_net_semanticmetadata_lire_imageanalysis_cnn_GetCNN_loadCNN 方法通过调用 Caffe 的模型操作相关接口实现对模型配置文件和模型参数文件的读取

和加载。

Java_net_semanticmetadata_lire_imageanalysis_cnn_GetCNN_getCNN 方法通过调用 Caffe 的模型测试相关接口，输入图片进行模型运算，从内存中读取运算结果，得到图像特征。

```
#include <string>
#include <vector>
#include <opencv2/opencv.hpp>
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>
#include <opencv2/imgproc/imgproc.hpp>
#include <opencv2/objdetect/objdetect.hpp>
#include <opencv2/imgproc/types_c.h>
#include <opencv2/objdetect/objdetect_c.h>
#include "boost/algorithm/string.hpp"
#include "caffe/caffe.hpp"
#include <jni.h>
#include "net_semanticmetadata_lire_imageanalysis_cnn_GetCNN.h"
using boost::shared_ptr;
using namespace std;
using namespace caffe;
using namespace cv;

#define dim1 3
#define dim2 224
#define dim3 224

boost::shared_ptr<Net<float>> feature_net;

JNIEXPORT void JNICALL Java_net_semanticmetadata_lire_imageanalysis_cnn_GetCNN_loadCNN
(JNIEnv *env, jobject obj, jstring NetTxt, jstring NetPara)
{
}

JNIEXPORT jfloatArray JNICALL Java_net_semanticmetadata_lire_imageanalysis_cnn_GetCNN_getCNN
(JNIEnv *env, jobject obj, jstring ImagePath, jstring layerName)
{
}
```

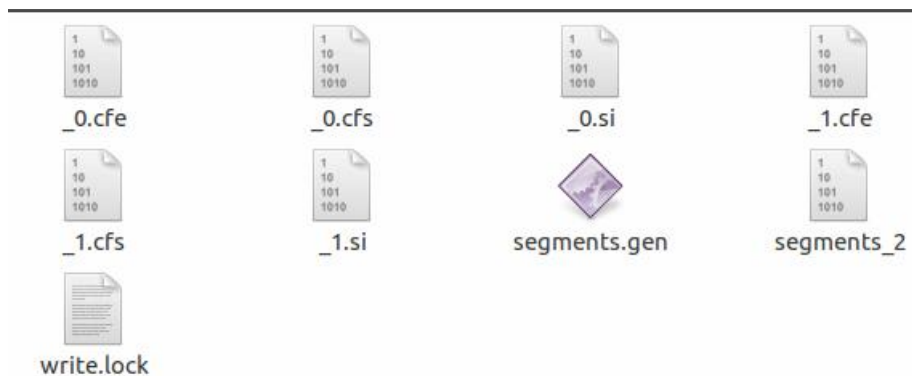
C++代码完成后，将 C++代码编译生成动态链接库，供 java 调用。

(3) 系统整合

负责人：刘少凡、宋昱材、吴沂楠、黄飞

将 Lire 代码与 C++代码生成的动态链接库进行整合，实现系统的 JNI 机制。并对系统进行简单的测试，确保运行成功。


入库成功后，设置的保存索引的文件夹目录下会出现新的索引文件，如图，其中_0开头的是第一次入库生成的索引，_1开头的是第二次入库生成的索引。





检索完成后，会依序显示检索结果，如图。


Search Result


Query Image :























number of pages : 6

Previous Page

Next Page

Switch To

1

Go

Current Page: 1