TECHNIQUE REPORT

**Approach to AI**

referringCNN



*The meaning of life is to give life a meaning …*

INDEX

[0 OVERVIEW 1](#_Toc5522811)

[0.0 Current Work 1](#_Toc5522812)

[0.0.1 Mission today 1](#_Toc5522813)

[0.0.2 Known issues 2](#_Toc5522814)

[0.1 Input Parameters 2](#_Toc5522815)

[0.1.1 How to input paramters 2](#_Toc5522816)

[0.1.2 Network parameters 2](#_Toc5522817)

[0.1.3 Layer parameters 3](#_Toc5522818)

[0.2 How to Make AI Distinguish Difficulty Levels 3](#_Toc5522819)

[1 PRELIMINARIES 4](#_Toc5522820)

[1.0 What is Referring Learning? 4](#_Toc5522821)

[1.1 Terminology 4](#_Toc5522822)

[1.1.1 Referring block 4](#_Toc5522823)

[2 DATA 5](#_Toc5522824)

[2.0 The Process of Data Management 5](#_Toc5522825)

[2.0.1 The format of training data 5](#_Toc5522826)

[2.0.2 Data preparation tools 5](#_Toc5522827)

[3 CODES IMPLEMENTATION 6](#_Toc5522828)

[3.0 The Architecture of referringCNN 6](#_Toc5522829)

[3.0.1 Displaying the Debuging Information 6](#_Toc5522830)

[3.0.2 Some Macros 6](#_Toc5522831)

[3.1 Main Working Flow 6](#_Toc5522832)

[3.2 Reusable Classes 6](#_Toc5522833)

[3.2.1 CInit 6](#_Toc5522834)

[4 GUI 7](#_Toc5522835)

[4.0 Parameters 7](#_Toc5522836)

[4.0.1 net config 7](#_Toc5522837)

[4.1 ini 7](#_Toc5522838)

[4.2 Message Manager 7](#_Toc5522839)

[4.2.1 progress bar 7](#_Toc5522840)

[4.3 Working Thread 7](#_Toc5522841)

[4.3.1 Job 8](#_Toc5522842)

[4.3.2 Job parameters 8](#_Toc5522843)

[4.4 Debug 8](#_Toc5522844)

[4.4.1 8](#_Toc5522845)

[5 ARTIFICIAL NEURAL NETWORK 9](#_Toc5522846)

[5.0 Toeplitz Matrix 9](#_Toc5522847)

[6 EXPERIMENTS 10](#_Toc5522848)

[6.0 The Architecture of Testing Flow 10](#_Toc5522849)

[6.1 Database Creating and Converting 10](#_Toc5522850)

[6.1.1 将图片集生成为训练用数据库 10](#_Toc5522851)

[6.2 Testcases 10](#_Toc5522852)

[6.2.1 Tiny testcase 10](#_Toc5522853)

[6.2.2 mnist 10](#_Toc5522854)

# OVERVIEW

At the beginning procedure of a training process, we fix some of the filter weights of classical CNN with classical convolution kernels such as sobel, kirch etc. After serveral fixing training phases when the net got stable, some of these fixing weights could be released free of fixing so that the network arrived at a refinement status.

Short term targets:

* 实现参考学习的轻量级平台
* 支持脚本进行网络参数匹配
* 测试平台在mnist上的训练速度和分类性能

Long range targets:

* The definition of referring learning and referring CNN
* The relationship between referring learning and transfer learning
* How to evaluating the robustness, stabling and generalization ability of a referring block

Network that can distinguish difficulty levels, easy cases will be trained lightly while difficult cases will be thouroughly exhausted trained in the network structure.

## Current Work

* Output the learnable parameters to a text file

level-training algorithm:

* in each level the training algorithm is at different stage with different strategy
* example: dark image vs bright image lead to different next processing stage

增加模板类 头文件：hybrid\_conv\_layer.h，实现：hybrid\_conv\_layer.hpp。

### Mission today

构造 tiny testcase 调试网络的构造流程

构建一个简单的深度网络进行测试

Tiny testcase - mnist

### Known issues

In this subsection, we collect all the known bugs to be fixed or problems to be clarified but currently not BIG enough to be solved immediately. Such bugs and problems will be solved later when we get time.

* CIni :: ReadInt & ReadText
* 未找到节的入口或参数的入口时直接报错（发现时间：2019年4月6日）
* 将来改为返回-1或NULL，并增加记录返回错误类型的error string

## Input Parameters

### How to input paramters

### Network parameters

* Learing rate：将来lr与lr\_decay可以考虑按照不同的layer设置不同的值
* Decay of learning rate: learning rate decreased in each training epoch
* See hyper\_network\_1.cpp, line 175

### Layer parameters

## How to Make AI Distinguish Difficulty Levels

# PRELIMINARIES

## What is Referring Learning?

## Terminology

### Referring block

# DATA

The classical flow of machine learning including the following steps:

1. Data source gathering
2. Row data processing
3. Feature fetching
4. Data classification
5. Model training
6. Model verification
7. Model testing
8. Prediction and classification

## The Process of Data Management

### The format of training data

### Data preparation tools

# CODES IMPLEMENTATION

## The Architecture of referringCNN

### Displaying the Debuging Information

### Some Macros

## Main Working Flow

## Reusable Classes

### CInit

# GUI

## Parameters

### net config

## ini

## Message Manager

### progress bar

## Working Thread

### Job

### Job parameters

## Debug

### 

# ARTIFICIAL NEURAL NETWORK

## Toeplitz Matrix

# EXPERIMENTS

* Observe the evolution procedure of the training convolution kernel, how could it become stable after serveral training rounds
* Output all the learnable parameters to a text file
* 先按原始算法训练，完成后将第一层的核referring为特征提取卷积核
* 第一层referring后，第二层referring第一层曾经训练好的核，以此初始化开始新的训练

## The Architecture of Testing Flow

## Database Creating and Converting

### 将图片集生成为训练用数据库

## Testcases

### Tiny testcase

### mnist