

XX论文

|  |  |
| --- | --- |
| 论文题目： | 基于嵌入式FPGA平台的高能效深度学习框架与应用 |
| 学 院： | 信息科学技术学院 |
| 年 级： | 2012级 |
| 专 业： | 计算机科学与技术 |
| 姓 名： | 赵睿哲 |
| 学 号： | 1200012778 |
| 指导教师： | 梁云 |

年 月 日

# 

# 摘要

2012年，基于深度学习的图像识别与分类算法取得了突破性的进展：在当年的ImageNet[[1]](#footnote-1)比赛中遥遥领先。自此，如何利用深度学习和神经网络算法，便成为计算机视觉领域的热门研究方向：基于图像的识别，分类，检索，以及基于动态视频流的“即时定位与地图重建”（Simultaneous Localization And Mapping, 简称SLAM）算法等等都获得性能与效果上的长足进步。但是这种方法有明显的局限性：传统深度学习算法往往基于大规模集群与GPU架构，而许多计算机视觉算法需要搭载于汽车、机器人乃至无人机等计算资源有限、能耗限制高且开发难度大的嵌入式设备上。因此，如何高效地实现与优化基于嵌入式平台的深度学习算法至关重要。

本研究实现了基于嵌入式FPGA平台的深度学习框架移植，并提出基于该平台开发深度学习应用的方法论。硬件平台选用赛灵思（Xilinx）公司推出的“全可编程”嵌入式SoC ZYNQ进行开发：双核ARM处理器与FPGA硬件在效率与能耗上均有优秀表现。开发平台使用Xilinx最新推出的SDSoC套件，极大地提高了开发效率。本研究将广泛应用于学术界和工业界的Caffe框架通过交叉编译部署于ZYNQ，并使用FPGA加速热点函数，相对于原始版本具有平均x倍的加速比。本研究最后选取基于卷积神经网络的立体匹配算法作为案例，以展现平台的计算效率与开发效率。

关键词

深度学习；ZYNQ；Caffe；SDSoC

# **Abstract**

Since when deep learning based image recognition algorithm

**Key words**

Key words;key words; key words(英文关键词内容必须与中文关键词完全对应。英文关键词采用Times New Roman小四号字书写，毕业论文、毕业设计行与行之间、段落和层次标题以及各段落之间均为1.5倍行距。关键词与关键词之间用“;”隔开)

目录

[摘要 I](#_Toc450657580)

[**Abstract** II](#_Toc450657581)

[前言 1](#_Toc450657582)

[第一章 基本原理 1](#_Toc450657583)

[1.1 深度学习 1](#_Toc450657584)

[1.1.1 深度学习算法 1](#_Toc450657585)

[1.1.2 卷积神经网络 2](#_Toc450657586)

[1.1.3 深度学习系统架构 2](#_Toc450657587)

[1.2 可编程逻辑门阵列（FPGA） 2](#_Toc450657588)

[1.2.1 架构概述 2](#_Toc450657589)

[1.2.2 高层次综合（HLS） 2](#_Toc450657590)

[1.3 片上系统（SoC） 2](#_Toc450657591)

[1.4 论文的主要工作 2](#_Toc450657592)

[第二章 总体设计 3](#_Toc450657593)

[2.1 深度学习框架Caffe 3](#_Toc450657594)

[2.1.1 软件架构 3](#_Toc450657595)

[2.1.2 GPU加速设计 3](#_Toc450657596)

[2.1.3 第三方库依赖 3](#_Toc450657597)

[2.1.4 使用方法 4](#_Toc450657598)

[2.2 ZYNQ平台 4](#_Toc450657599)

[2.2.1 系统架构 4](#_Toc450657600)

[2.2.2 ARM处理器 4](#_Toc450657601)

[2.2.3 FPGA硬件 4](#_Toc450657602)

[2.2.4 Zedboard开发板卡 4](#_Toc450657603)

[2.3 SDSoC开发工具 4](#_Toc450657604)

[2.3.1 基本原理与组成 4](#_Toc450657605)

[2.3.2 工作流程 4](#_Toc450657606)

[2.4 综合设计方案 5](#_Toc450657607)

[第三章 主体工作 5](#_Toc450657608)

[3.1 系统架构设计 5](#_Toc450657609)

[3.2 FPGA加速器设计 5](#_Toc450657610)

[3.2.1 优化原则 5](#_Toc450657611)

[3.2.2 GEMM算法优化实现 6](#_Toc450657612)

[3.2.3 GEMV算法优化实现 6](#_Toc450657613)

[3.3 ARM处理系统与数据通路设计 6](#_Toc450657614)

[3.3.1 ARM处理系统设计 6](#_Toc450657615)

[3.3.2 数据通路设计 6](#_Toc450657616)

[3.4 应用：基于卷积神经网络的立体匹配算法实现 6](#_Toc450657617)

[3.4.1 立体匹配算法 6](#_Toc450657618)

[3.4.2 神经网络设计 6](#_Toc450657619)

[3.4.3 前处理与后处理 6](#_Toc450657620)

[第四章 测试结果 7](#_Toc450657621)

[4.1 FPGA加速器测试 7](#_Toc450657622)

[4.1.1 GEMM性能测试 7](#_Toc450657623)

[4.1.2 GEMV性能测试 7](#_Toc450657624)

[4.1.3 综合性能测试 7](#_Toc450657625)

[4.1.4 资源利用 8](#_Toc450657626)

[4.2 Caffe整体性能测试 8](#_Toc450657627)

[4.2.1 单元测试 8](#_Toc450657628)

[4.2.2 样例应用测试 8](#_Toc450657629)

[4.3 立体匹配算法效率测试 8](#_Toc450657630)

[第五章 总结与展望 9](#_Toc450657631)

[5.1 主要工作总结 9](#_Toc450657632)

[5.2 未来优化方向 9](#_Toc450657633)

[结论 10](#_Toc450657634)

[参考文献 11](#_Toc450657635)

[附录一 12](#_Toc450657636)

[致谢 13](#_Toc450657637)

（备注：目录按2～3级标题编写“目录”二字使用黑体小二号字居中书写，隔行书写目录内容，“摘要、Abstract、正文的一级标题、结论、参考文献、附录目录、致谢”采用黑体小四号字书写，正文其他层次标题均采用宋体小四号字书写；“摘要、Abstract”与“正文”之间隔一行；正文中的二级标题、三级标题相对于上一级标题均缩进二个空格书写；目录行与行之间均为1.5倍行距；目录内容多者，正文中的二、三级标题可使用宋体五号字书写，行与行之间可采用单倍行距。目录修改时单击目录点右键选择更新域，选择更新整个目录。之后，在“Abstract”的页码后回车加一空行。）

# 前言

正文采用宋体（英语用Times New Roman）小四号字，毕业论文、毕业设计行与行之间、段落和层次标题以及各段落之间均为1.5倍行距。[[2]](#footnote-2)

毕业论文的前言应综合评述前人工作，说明论文工作的选题目的、背景和意义、国内外文献综述以及论文所要研究的主要内容，对所研究问题的认识，以及提出问题等。前言只是文章的开头，可不写章号，也可不出现“前言”二字。

毕业设计的前言部分应说明设计的目的、意义、范围及应达到的技术要求；简述课题在国内外的发展概况及存在的问题；阐明设计的指导思想；阐述设计应解决的主要问题。[[3]](#footnote-3)

# 第一章 基本原理

正文采用宋体（英语用Times New Roman）小四号字，毕业论文、毕业设计行与行之间、段落和层次标题以及各段落之间均为1.5倍行距。

## 1.1 深度学习

正文采用宋体（英语用Times New Roman）小四号字，毕业论文、毕业设计行与行之间、段落和层次标题以及各段落之间均为1.5倍行距。

具体内容 具体内容 具体内容 具体内容 具体内容 具体内容 具体内容[[[4]](#endnote-1)]。

### 1.1.1 深度学习算法

正文采用宋体（英语用Times New Roman）小四号字，毕业论文、毕业设计行与行之间、段落和层次标题以及各段落之间均为1.5倍行距。

### 1.1.2 卷积神经网络

正文

### 1.1.3 深度学习系统架构

正文

## 1.2 可编程逻辑门阵列（FPGA）

正文

### 1.2.1 架构概述

正文

### 1.2.2 高层次综合（HLS）

正文

## 1.3 片上系统（SoC）

正文

## 1.4 论文的主要工作

正文

# 第二章 总体设计

正文

## 2.1 深度学习框架Caffe

正文

### 2.1.1 软件架构

正文

### 2.1.2 GPU加速设计

正文

### 2.1.3 第三方库依赖

正文

### 2.1.4 使用方法

正文

## 2.2 ZYNQ平台

正文

### 2.2.1 系统架构

正文

### 2.2.2 ARM处理器

正文

### 2.2.3 FPGA硬件

正文

### 2.2.4 Zedboard开发板卡

正文

## 2.3 SDSoC开发工具

正文

### 2.3.1 基本原理与组成

正文

### 2.3.2 工作流程

正文

## 2.4 综合设计方案

正文

# 第三章 主体工作

正文

## 3.1 系统架构设计

正文

## 3.2 FPGA加速器设计

正文

### 3.2.1 优化原则

正文

### 3.2.2 GEMM算法优化实现

正文

### 3.2.3 GEMV算法优化实现

正文

## 3.3 ARM处理系统与数据通路设计

正文

### 3.3.1 ARM处理系统设计

正文

### 3.3.2 数据通路设计

正文

## 3.4 应用：基于卷积神经网络的立体匹配算法实现

正文

### 3.4.1 立体匹配算法

正文

### 3.4.2 神经网络设计

正文

### 3.4.3 前处理与后处理

正文

# 第四章 测试结果

正文

## 4.1 FPGA加速器测试

正文

### 4.1.1 GEMM性能测试

正文

### 4.1.2 GEMV性能测试

正文

### 4.1.3 综合性能测试

正文

### 4.1.4 资源利用

正文

## 4.2 Caffe整体性能测试

正文

### 4.2.1 单元测试

正文

### 4.2.2 样例应用测试

正文

## 4.3 立体匹配算法效率测试

正文

# 第五章 总结与展望

正文

## 5.1 主要工作总结

正文

## 5.2 未来优化方向

正文

# 结论

毕业论文的结论是对整个论文主要成果的归纳，应突出论文的创新点，以简练的文字对论文的主要工作进行评价。若不可能得出应有的结论，则需进行必要的讨论。可以在结论或讨论中提出建议、研究设想及尚待解决的问题等。

毕业设计的结论是概括说明设计的情况和价值，分析其优点和特色，有何创新，性能达到何水平，并应指出其中存在的问题和今后改进的方向。

结论要单独成页，“结论”二字采用黑体小二号字居中书写，隔行采用宋体（英语用Times New Roman）小四号字书写具体内容，行与行之间、各段落之间均为1.5倍行距。

# 参考文献

# 附录一

附录序号采用“附录一”、“附录二”、“附录1”、“附录2” 、“附录A”、“附录B”等，用四号黑体字左起顶格排写，其后不加标点符号，空一行书写附录内容。附录内容采用宋体（英语用Times New Roman）小四号字书写。行与行之间、各段落之间均为1.5倍行距。

附录可有可无，当无附录内容时，此页可删除；当附录唯一时，只写“附录”即可。

# 致谢

致谢要单独成页，“致谢”二字采用黑体小二号字居中书写，隔行采用宋体（英语用Times New Roman）小四号字书写具体内容。行与行之间、各段落之间均为1.5倍行距。

1. ImageNet 2012 [↑](#footnote-ref-1)
2. 依次书写作者名、资料的篇名、发表的刊物名、出版年份和期号。 [↑](#footnote-ref-2)
3. 如果是著作则应写明出版单位和出版年份，见《黑龙江大学本科生毕业论文（设计）撰写规范》。 [↑](#footnote-ref-3)
4. [] 主要责任者.文献题名[J].刊名.出版年份,卷号(期号):起止页码． [↑](#endnote-ref-1)