



宁波大学
NINGBO UNIVERSITY

本科毕业设计（论文）

题目：某种设计与实现

Design and Implementation of something

学 院	信息科学与工程学院
专 业	电气工程及其自动化
班 级	19 自动化班
学 号	196000001
学生姓名	路某
指导教师	我导
完成日期	2023.04.26

诚信承诺

我谨在此承诺：本人所写的毕业论文《某种设计与实现》的主体均系本人独立完成，没有抄袭行为，凡涉及其他作者的观点和材料，均作了注释，若有不实，后果由本人承担并愿接受校方的处分。

承诺人（签名）：

年 月 日

目 录

摘 要	I
Abstract	II
1 背景与意义	1
1.1 选题背景	1
1.2 相关研究工作	1
1.2.1 基于插值法的图像超分辨率重建算法	1
1.2.2 基于成像模型重建的超分辨率算法	1
1.3 论文结构安排	2
2 图像超分辨率算法定义	3
2.1 超分辨率重建与生成对抗策略	3
参考文献	5
致 谢	6
附录 A 补充内容	7

某种设计与实现

摘 要

图像超分辨率重建一直是计算机视觉的一个重要研究方向，近年来生成对抗网络在计算机视觉领域展现了强大的潜力。……………。

关键字 超分辨率重建；生成对抗网络；双卷积神经网络；人眼视觉系统；全参考图像质量评价

Design and Implementation of something

Abstract

The image super-resolution is always a key domain of computer vision. In recent years, the Generative Adversarial Network (GAN) has shown a great potentiality in this area. …….

KEYWORDS Super-Resolution Reconstruction; GAN; Dual CNN; HVS; FR-IQA

1 背景与意义

1.1 选题背景

图像超分辨率 (SR)^[1] 是一种在计算机视觉领域的图像生成技术，是一种将现有图像通过一定的方式提升分辨率大小的计算过程。提高分辨率能够让图像包含更多的信息，在医疗、工业、商业等诸多领域都有广泛的应用前景。

.....

.....

1.2 相关研究工作

自 Keys R. 于 1981 年提出“双三次插值法”^[2] 以来，图像超分辨率重建一直是计算机图像领域的一个研究热点。目前，超分辨率图像重建工作主要沿着插值法、成像模型重建和机器学习这三大技术方向进行实现。同时，为了设计一个基于真实度的图像评价模型，本节还将介绍几种经典的全参考图像质量评价算法。

1.2.1 基于插值法的图像超分辨率重建算法

基于插值法的图像超分辨率算法主要针对已有的输入数据进行工作，通过对临近像素信息的获取和分析，拟合出能够填入临近像素间的新像素的成像信息。在保证图像整体风格不变的基础上，该方法能提高图像分辨率，增强图像可用性和可识别性。

.....

.....

1.2.2 基于成像模型重建的超分辨率算法

基于重建模型的超分辨率算法通过将图像的一些基本概念和成像规律作为先验加入到超分辨率重建过程中，使得图像的超分辨率重建从一个随机性问题变为有先验约束的求确定解的问题。

.....

.....

基于重建模型的超分辨率算法通过将图像的一些基本概念和成像规律作为先验加入到超分辨率重建过程中，使得图像的超分辨率重建从一个随机性问题变为有先验约束的求确定解的问题。

基于重建模型的超分辨率算法通过将图像的一些基本概念和成像规律作为先验加入

到超分辨率重建过程中，使得图像的超分辨率重建从一个随机性问题变为有先验约束的求确定解的问题。

1.3 论文结构安排

本文共五章，论文后续内容的结构安排如下：

相关理论知识（第二章）

基于对抗生成的图像超分辨率算法（第三章）

基于人眼视觉感知的图像评价模型（第四章）

总结和展望（第五章）

第二章介绍了 SRGAN 算法提出的动机和背景以及具体的算法设计，并对网络的主要工作原理进行了分析。另外，介绍了两种生成对抗模型 ESRGAN 和 WGAN，概括说明了两种模型的创新点和部分结构。其中详细介绍了 Wasserstein 距离并对带有 Wasserstein 距离的感知函数单元进行了推导和功能分析。最后结合公式和图像介绍了几种经典的图像评价指标，包括 MSE 值、PSNR 值、SSIM 值以及主观真实性评价等。

第三章首先分析了当下对抗生成网络存在的问题。……

第四章提出了一种新的基于人眼视觉感知的图像评价算法，用来弥补如今超分辨率重建领域基于真实性的评价算法较为欠缺的问题。本文通过增加融合亮度层提高模型对失真图像缺失边缘的感知，减弱传统梯度算法对平滑区域过于强调的效果。……

第五章对全文工作进行总结的同时，也对超分辨率重建模型的优化方向和图像质量评价算法的优化路线方面做了展望。

2 图像超分辨率算法定义

单幅图像超分辨率算法是一种利用一幅经过下采样、反卷积等“退化”过程后的低分辨率图像经过一系列的计算方法得到对应高分辨率图像的计算过程，图像退化计算可以建模为：

$$I_L = D(I_H; \mu) \quad (2.1)$$



图 2-1 示例图片

图像超分辨率计算过程可建模如(2.1)



图 2-2 示例图片 2



图 2-3 示例图片 3

其中， I_L 为低分辨率图像， I_H 为与之对应的高分辨率图像， D 代表一种图像退化算法， μ 为退化因子，为经超分算法得到的高分辨率图像， S 代表一种超分辨率算法， v 为超分因子。

2.1 超分辨率重建与生成对抗策略

生成对抗策略是近几年来深度学习领域研究者重点研究方向。通过生成模型与判别器模型的对抗实现从输入源数据自身概率分布到目标数据的概率分布之间的函数映射参数的获取，从而获得一个源数据到目标数据的映射参数网络，这样相互迭代式的无监督学习方法提高了网络学习效率。……。经典的生成对抗网络模型如下表示 [26]：

生成对抗策略是近几年来深度学习领域研究者重点研究方向。通过生成模型与判别器模型的对抗实现从输入源数据自身概率分布到目标数据的概率分布之间的函数映射参数的获取，从而获得一个源数据到目标数据的映射参数网络，这样相互迭代式的无监督学习

方法提高了网络学习效率。……。经典的生成对抗网络模型如下表示 [26]:

$$\int f(x) dx \quad (2.2)$$

生成对抗策略是近几年来深度学习领域研究者重点研究方向。通过生成模型与判别器模型的对抗实现从输入源数据自身概率分布到目标数据的概率分布之间的函数映射参数的获取,从而获得一个源数据到目标数据的映射参数网络,这样相互迭代式的无监督学习方法提高了网络学习效率。……。经典的生成对抗网络模型如下表示 [26]:

表 2-1 数据库间内容对比

数据库	原始图像数	失真图像数	失真类型	主观评价法	参评人员数
LIVE	29	779	5	DMOS	161
TID2013	25	3000	24	MOS	971

参考文献

- [1] Kim D H, Ko K W, Park C. CSMA/CD with Reservations in Wireless Communication for Distributed Robot System[J]. International Journal of Control, Automation and Systems, 2010, 8(3): 583-596 [2023-01-10].
- [2] 宋奇. 华为 HarmonyOS 2.0 不一样的体验[J]. 计算机与网络, 2020, 46(18): 72.

致 谢

衷心感谢导师 ××× 教授和物理系 ×× 副教授对本人的精心指导。他们的言传身教将使我终生受益。

在美国麻省理工学院化学系进行九个月的合作研究期间，承蒙 ××× 教授热心指导与帮助，不胜感激。

感谢 ××××× 实验室主任 ××× 教授，以及实验室全体老师和同窗们学的热情帮助和支持！

本课题承蒙国家自然科学基金资助，特此致谢。

附录 A 补充内容

附录是与论文内容密切相关、但编入正文又影响整篇论文编排的条理和逻辑性的资料，例如某些重要的数据表格、计算程序、统计表等，是论文主体的补充内容，可根据需要设置。



图 A-1 fig-appendix

以下为分图示例

以下为分图示例

以下为分图示例

以下为分图示例

以下为分图示例：



图 A-2 多个分图的示例

以下为表格示例

以下为表格示例

以下为表格示例

以下为表格示例

以下为表格示例

以下为表格示例

以下为表格示例：

表 A-1 table-example-appendix

数据库	原始图像数	失真图像数	失真类型	主观评价法	参评人员数
LIVE	29	779	5	DMOS	161
TID2013	25	3000	24	MOS	971

以下为代码示例

以下为代码示例

以下为代码示例

以下为代码示例 以下为代码示例

以下为代码示例：

```
1  /**
2   * @brief  LED 亮度控制中断
3   * @param  none
4   * @retval none
5   */
6  typedef uint8_t LED8BitsType;
7  void LED_PeriodElapsedCallback(void){
8      int i = 0;
9      static LED8BitsType led_bright_tick = 0;
10     led_bright_tick += 1;
11     if(led_bright_tick>=100){
12         led_bright_tick = 0;
13     }
14     if(led_bright_tick == led.brightness){
15         HAL_GPIO_WritePin(FnLED_GPIO_Port, FnLED_Pin, GPIO_PIN_RESET);
16     }else if(led_bright_tick == 0){
17         HAL_GPIO_WritePin(FnLED_GPIO_Port, FnLED_Pin, GPIO_PIN_RESET);
18     }
19 }
```

代码 A-1 code-example