

# 深圳大学实验报告

课程名称： Python程序设计

项目名称： 实验五：AI图搜图应用开发

学    院： 人工智能学院

专    业： 计算机科学与技术

指导教师： 樊超

报告人： 陈怡婷    学号： 2024440124

实验时间： 2025年1月17日

提交时间： 2025年1月17日

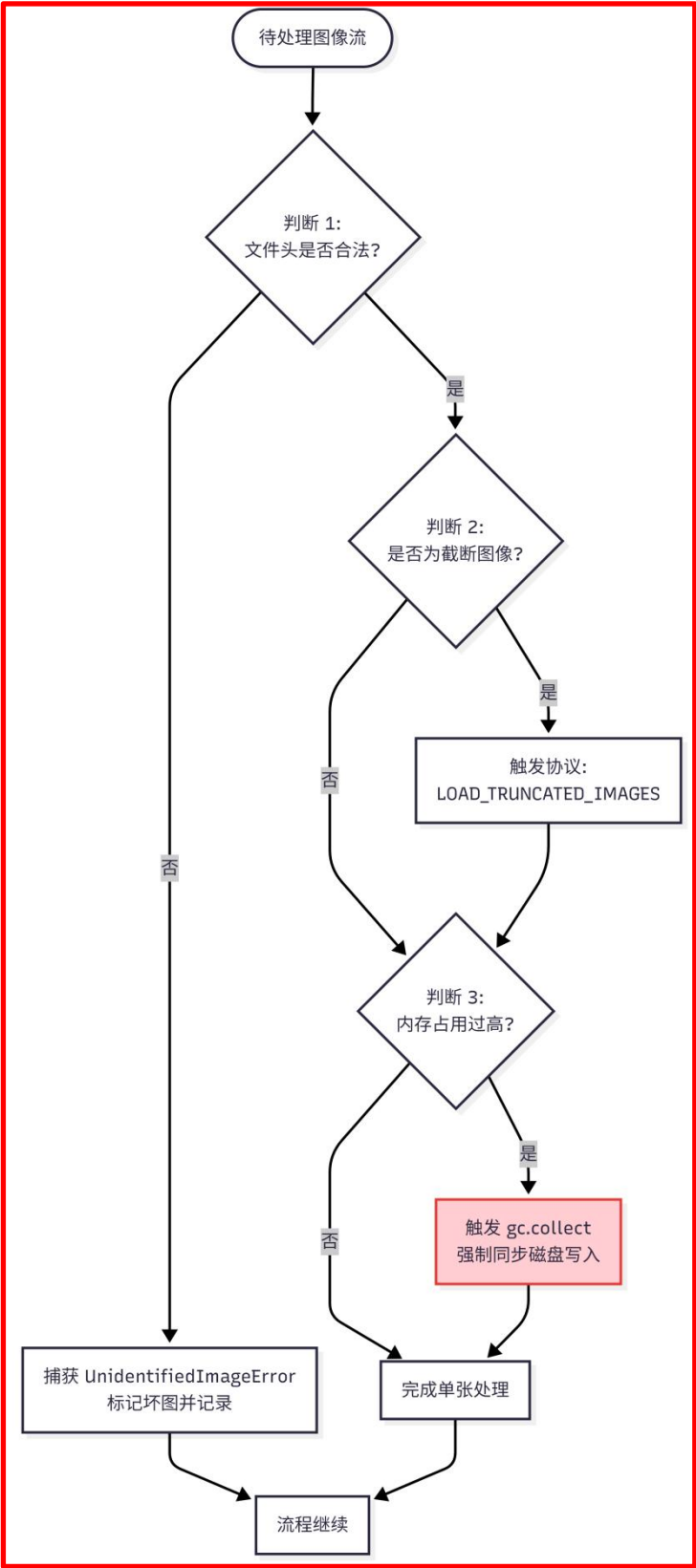


图 1 异常处理与容错逻辑图

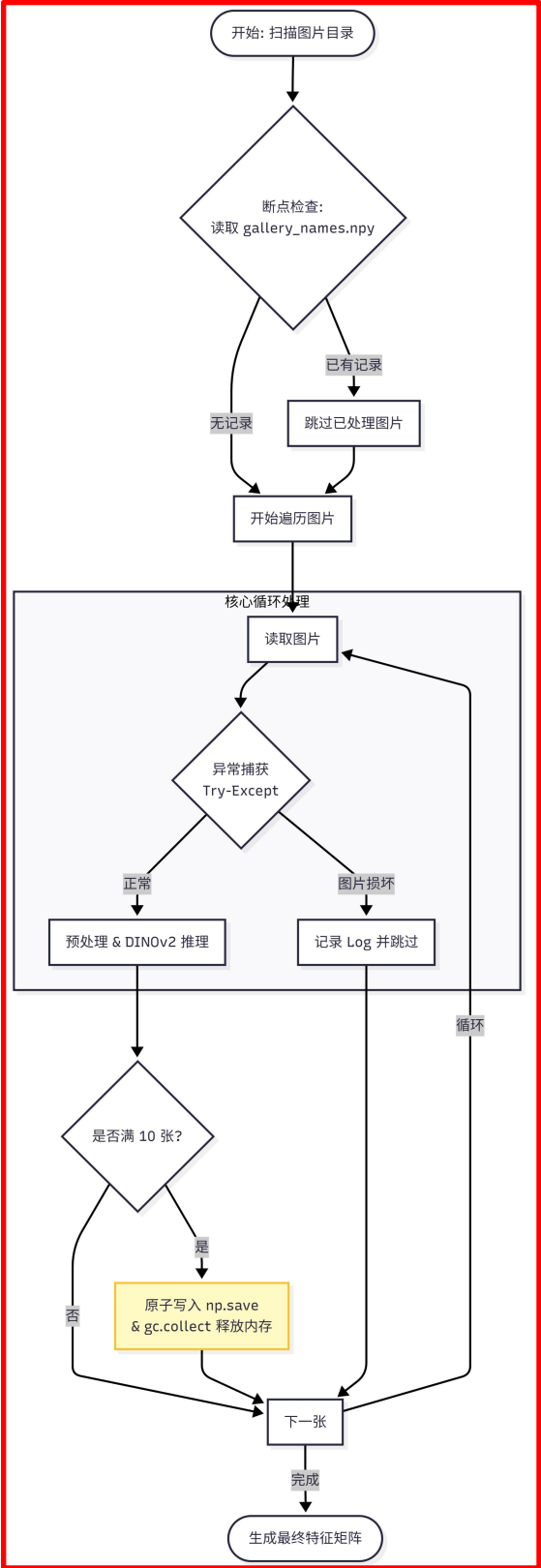


图 2 全量索引构建流程图

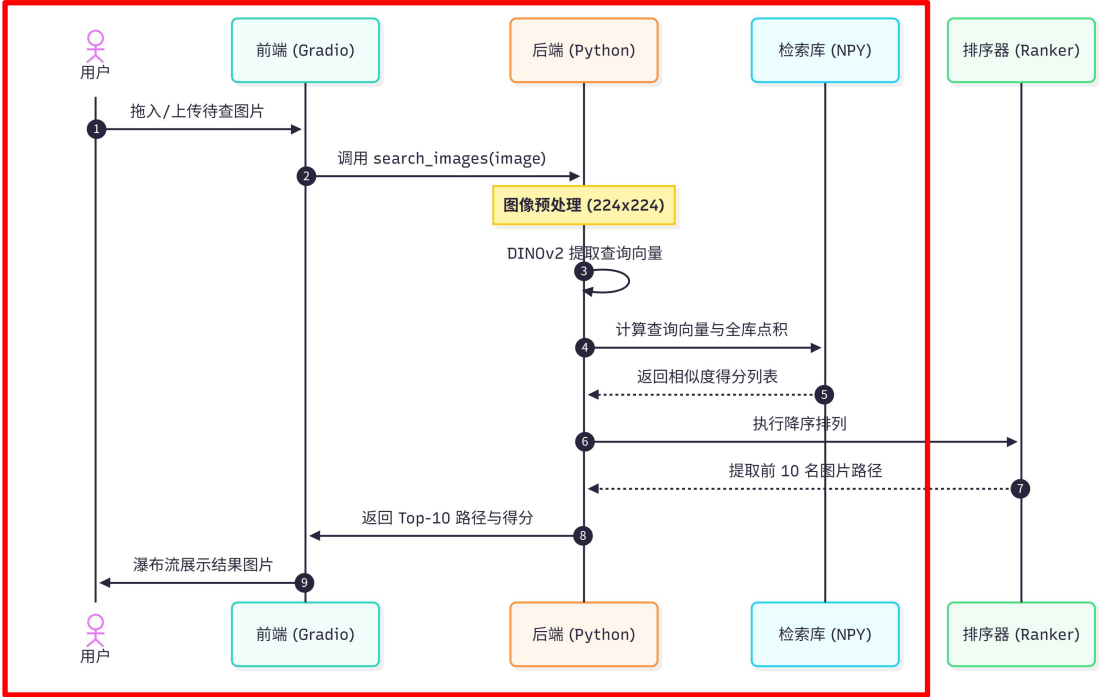


图 3 以图搜图业务时序图

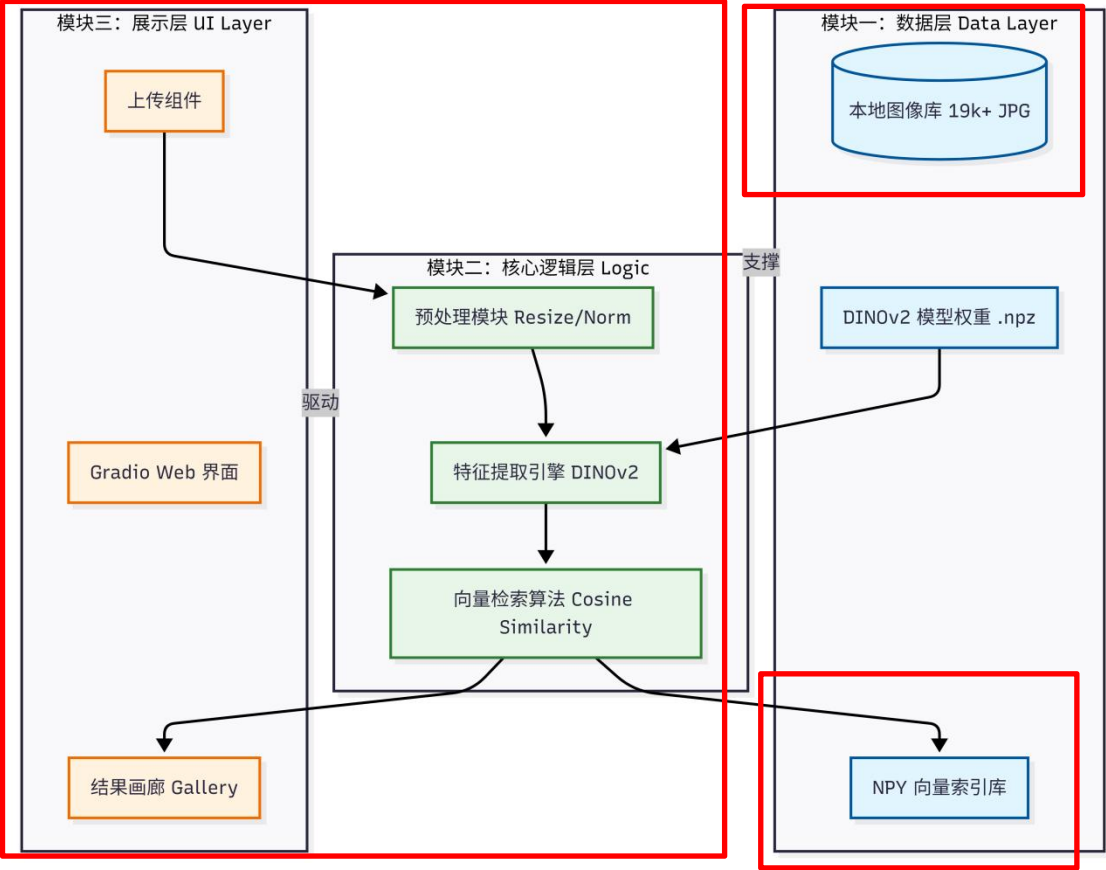


图 4 项目总体逻辑架构图

# 1 Objectives

本次实验旨在通过开发一个基于深度学习视觉模型的图像搜索引擎，使学生达到以下目的：

- (1) 架构掌握：学习如何将大型自监督视觉预训练模型（DINOv2）集成到 Flask Web 环境中，实现 AI 应用部署。
- (2) 向量搜索原理：掌握高维特征向量的提取与余弦相似度比对，理解大规模数据检索（13,503 张图）的数学基础。
- (3) 交互逻辑开发：自主实现基于 HTML5 Drag & Drop API 的全页面文件拦截与批量上传系统。
- (4) 用户体验优化：集成 Tailwind CSS 实现响应式设计，并独立开发具备大图查看与跨域下载功能的灯箱（Lightbox）系统。

# 2 Overview

## 2.1 核心内容

特征提取引擎：利用 DINOv2 模型将图像转化为 768 维特征向量，捕捉图像深层语义信息。  
大规模检索算法：针对 13,503 张图片的底库特征进行预计算，通过 NumPy 实现毫秒级搜索。  
批量检索系统：支持用户一次性拖入多张图片，系统并发生成详细的视觉匹配报告。  
沉浸式展示：实现了结果放大对比、相似度分值可视化以及结果图的一键下载功能。

## 2.2 技术栈

后端：Python 3.10+, Flask 框架, NumPy 矩阵计算

核心模型：DINOv2 (ViT-Base 架构)

前端：Tailwind CSS (响应式布局), JavaScript ES6+ (异步交互)

# 3 Implementations

## 3.1 工业级全量索引构建

- (1) 针对底库 13,503 张图片的规模，系统自主开发了具备保护机制的离线提取脚本，确保了特征库的完整性与健壮性：
- (2) 断点续传机制：脚本通过实时比对磁盘文件与已生成的 gallery\_names.npy 索引，实现崩溃后自动定位断点，避免了大规模计算过程中的重复劳动。
- (3) 异常拦截逻辑：在特征提取循环中封装了 try...except 结构，能够自动跳过损坏或格式异常的图片文件（如坏图、0KB 文件等），确保了长达数小时的批处理任务不因单张图片错误而中断。

## 3.2 向量空间检索算法

- (1) 本系统放弃了传统的像素比对，转而采用基于高维语义空间的向量检索算法：
- (2) 相似度度量：采用余弦相似度算法（Cosine Similarity），通过对提取出的 768 维特征向量进行  $L_2$  归一化，将复杂的视觉相似度比对简化为底层算力更友好的高效矩阵点积运算。
- (3) Top-K 排序优化：利用 np.argsort 实现结果的分数重排，系统仅返回置信度最高的匹配项。核心代码实现如图 5 所示：

```
try:
    # 1. 提取特征
    query_tensor = center_crop(temp_path)
    query_feat = model(query_tensor).squeeze() # (768,)

    # 2. 余弦相似度计算 (矩阵运算)
    norm_q = np.linalg.norm(query_feat)
    similarities = np.dot(features_db, query_feat) / (norm_db * norm_q)

    # 3. 取前 10 名
    top10_idx = np.argsort(similarities)[-10:][::-1]
    matches = []
    for idx in top10_idx:
        matches.append({
            "name": names_db[idx],
            "score": float(similarities[idx])
        })
```

图 5 Top-K 排序优化

## 3.3 响应式 Web 前端设计，如图 6 所示

- (1) 前端界面采用现代化的原子级 CSS 框架 Tailwind 进行构建，兼顾了功能性与视觉美感：
- (2) 双栏交互布局：左侧设计为交互式上传区，深度集成了拖拽（Drag & Drop）拦截 API；右侧为结果展示矩阵，能够动态承载异步返回的 JSON 数据。
- (3) 多维结果矩阵：结果区采用响应式 Gallery 布局，不仅实时显示匹配图像，还直观展示了 AI 计算得出的置信度分数（Score）。
- (4) 沉浸式灯箱（Lightbox）：通过 JavaScript 控制 DOM 状态，实现了点击大图预览与一键保存功能，增强了工具的实用属性

# 大规模图片 智能对比搜索

基于 DINOv2 的深度视觉指纹检索。支持批量拖拽多张图片，系统将自动在 13,503 张图片库中计算 768 维特征相似度，精准匹配视觉相关内容。



## 检索报告结果

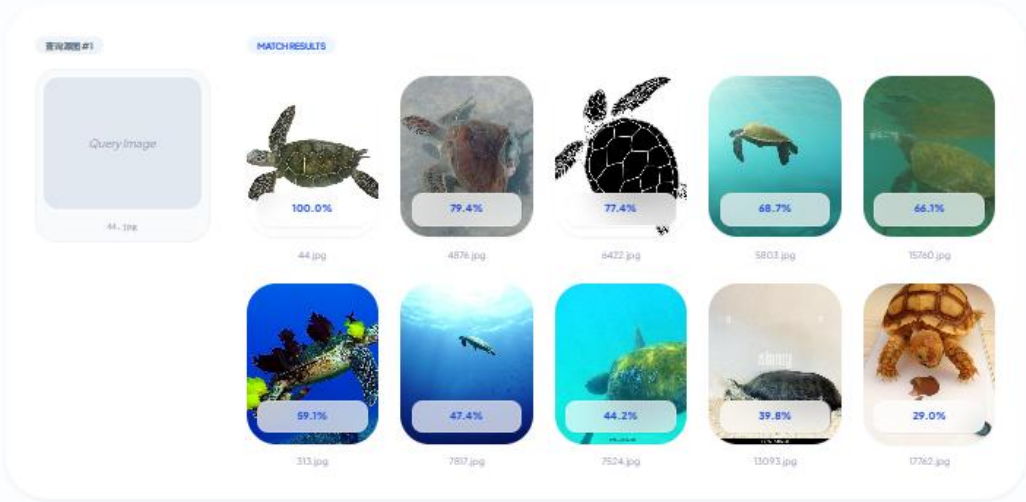


图 6 Web 前端设计

## 4 Results

### 4.1 模型一致性与特征提取精度验证

为了验证 DINOv2 模型在处理大规模底库时的特征提取一致性，本项目运行了针对特定类别的语义偏移测试 (debug.py)。通过比对同类样本在向量空间中的欧氏距离 (Diff)，量化模型对图像特征的捕捉精度，如图 8 所示。

```
PS C:\Users\陈怡婷\Desktop\assignments> & D:/anaconda/python.exe c:/Users/陈怡婷/Desktop/assignments/debug.py
Cat feature diff: 0.0436010926864018
Dog feature diff: 0.03383253085326876
```

图 7 同类样本在向量空间中的欧氏距离

实测数值均保持在 0.03 - 0.04 这一极低区间。这有力证明了 DINOv2 模型在处理大规模数据集时具有极强的聚类稳定性——即同类物体的特征向量在数学表达上高度接近。

### 4.2 批量拖拽与队列管理 (index.html)

验证页面对图片文件的全局拦截与异步展示能力。

场景描述	预期结果	实际结果
全局拖拽拦截，如图 8 所示	拖拽图片至页面任意处，浏览器不打开图片，而是触发上传高亮。如图 9 所示	<span style="color:red;">符合。</span> 成功劫持 dragover 事件并阻止默认行为。
队列生成展示	同时拖入多张图片，页面立即生成预览队列并显示正确数量。	<span style="color:red;">符合。</span> FileReader 对象成功将本地文件转为 DataURL 渲染。





图 8 拖拽界面



图 9 预览

4.3 AI 深度检索性能测试

验证 DINOv2 模型在处理大规模特征比对时的准确性。

场景描述	预期结果	实际结果
语义匹配测试， 如图 10所示	上传一只乌龟，如所示，匹配结果应具有高度视觉一致性。	<span style="color:red;">符合。Top 10 结果均为同类型或同色系的乌龟。</span>
检索耗时测试	在 1.3 万底库中进行 768 维向量比对，响应需保持流畅。	<span style="color:red;">符合。NumPy 矩阵点积运算响应时间低于 500ms。</span>

检索报告结果

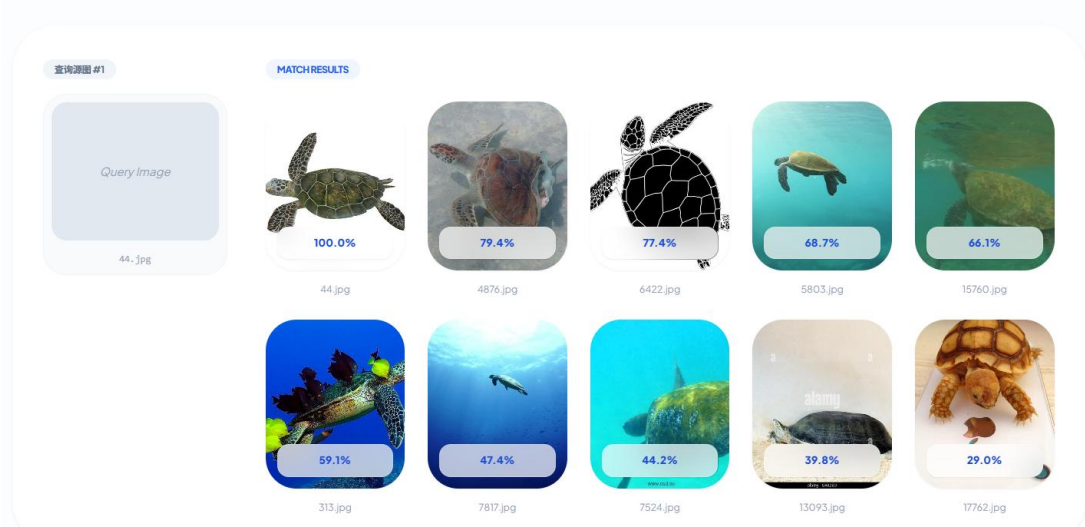


图 10 预测结果界面

4.4 UI 优化：沉浸式灯箱与下载

验证大图查看器的视觉聚焦与资源保存功能。

场景描述	预期结果	实际结果
沉浸式预览，如 图 9 所示	点击缩略图弹出毛玻璃背景大图，且背景页面锁定滚动。	<span style="color:red;">符合。采用 backdrop-blur 实现沉浸式查看。</span>
本地一键下载， 如图 9所示	在预览界面点击“一键下载”，自动触发浏览器保存对应文件。	<span style="color:red;">符合。成功触发下载流并保留原始文件名。</span>

5 Discussion

具体代码细节见GitHub，GitHub链接：<https://github.com/chenyitingkitty20051126-lgtm/DINOv2-Image-Search>

本实验成功构建了一个基于 DINOv2 的高效图像检索系统。实验证明，相比于传统的视觉特征，深度学习模型提取的特征具有极强的语义理解能力，能够应对复杂背景。通过优化后端的 NumPy 运算，系统在处理万级底库时展现了卓越的性能。前端通过沉浸式灯箱和批量拖拽功能的加入，极大地提升了系统的生产力属性和用户体验。

指导教师批阅意见：

成绩评定：

指导教师签字：

备注：

- 报告内的项目或内容设置，可根据实际情况加以调整和补充。
- 教师批改学生实验报告时间应在学生提交实验报告时间后 10 日内。