

# 图片搜索引擎 ImgSE 项目报告

## 《搜索引擎技术基础》课程设计

任自厚

2022 年 5 月 28 日

## 1 问题描述

搭建一个前后端完备的图片搜索引擎。核心功能为图片检索，拓展功能包括条件筛选、以图搜图、查询纠错、跨语言、跨数据集等。

受限于数据集提供信息等，本项目实现了如下功能：

- 通过文本检索图片，按相关性返回结果
- 以标签、主导色、图片尺寸为条件筛选搜索结果
- 通过图片检索图片，按相似度返回结果
- 将搜索结果分页返回
- 完整支持以上功能的浏览器前端界面
- 不受限于特定数据集的数据插入接口，支持轻松扩充图库

在 demo 版本当中，使用的图片集为谷歌开放数据集中的 F 子集，由于该数据集提供的图片主题覆盖面较广，相似图片较少，我在图库中手动添加了来自互联网上的 10 张海滩图片和 10 张小鸟图片，用作以图搜图功能的测试。图库中图片总数为 95,578。

## 2 实现模块与关键功能

### 2.1 后端：搜索引擎

搜索引擎具有两个核心功能（文本搜图、以图搜图）和一个辅助功能（条件筛选）。搜索流程如图 1 所示。

#### 2.1.1 文本搜图

谷歌开放数据集中，每张图片已经被人工标注了描述文本，因此“文本搜图”这一功能等价于一个简易的文本搜索引擎，其核心流程包括倒排索引和相关性排序。

为了使得倒排索引内容更精准、贴合搜索意图，在将图片的文本描述写入倒排索引前需要先经过文本预处理。文本预处理删除了文本中的停用词、标点符号等，并统一了字母大小写。本项目使用 Python 内置 `Dict[List[str]]` 类型作为倒排索引，实验证明在当前载入的图片量级下，字典类型也能很好的完成索引任务。

相关性排序依据为 BM25，在具体实现上我借助了第三方库 `rank_bm25`。

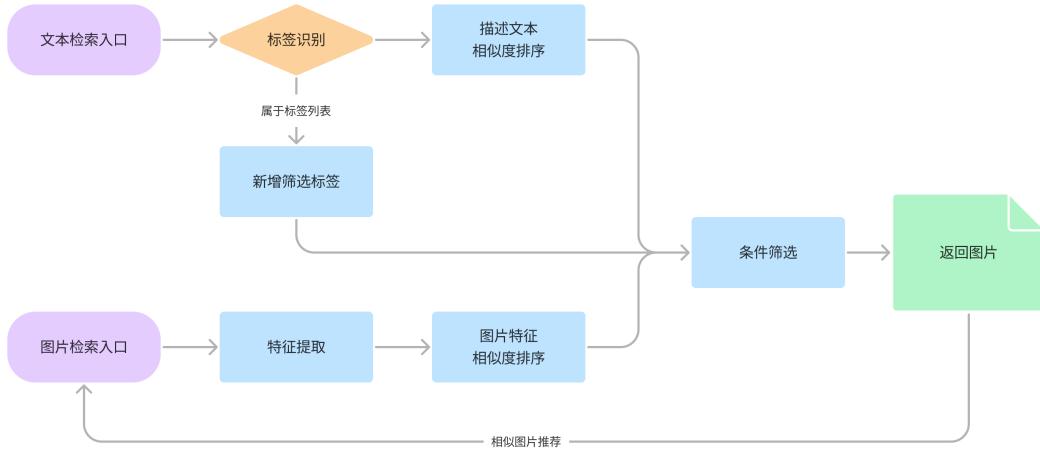


图 1: 搜索引擎工作流程

### 2.1.2 以图搜图

以图搜图算法的核心流程为提取图片特征、计算特征相似度。

图片相似度是一个很广义的概念，可以是视觉相似度（长得像，如图片变换）、标签相似度（同一抽象物体，如猫、狗等）、主体相似度（同一具体物体，如一栋建筑不同视角）等等。因此，图片特征的提取也可以从多个方面入手，如色彩特征、轮廓特征等；近年来使用 CNN 提取特征成为了主流。受限于电脑容量与算力，并考虑到搜索引擎需要快速提取特征并进行相似度计算，本项目选择使用基于区域的色彩直方图作为图片特征，实现了一个简易的以图搜图引擎。在具体实现上，该算法将每张图片分为左上、右上、左下、右下、中心五个区域，并分别计算每个区域中亮度 (L)、色相 (H)、饱和度 (S) 的分布直方图。将五个区域的分布直方图拼接在一起，即形成该图片的特征向量。

在特征相似度计算上，我使用 Ball-Tree 作为搜索数据结构。相比于平凡 KD-Tree，Ball-Tree 在高维向量相似度计算上表现突出，这正适用于高维图片特征向量。sklearn 库中已经提供了相关接口，直接调用即可。

### 2.1.3 条件筛选

在本项目中，条件筛选功能位于检索功能的下游，即搜索引擎会通过先检索出全部<sup>1</sup>符合描述的图片，再调用筛选算法，对图片标签、尺寸（像素数）、主导色进行筛选，将符合的结果返回。

在具体筛选算法上，由于图片标签较常用，且命中率较低（一个标签的图片数远小于全集图片数），使用倒排索引的方法，判断目标图片是否在相应索引列表（多个标签取交集）中；尺寸和主导色的筛选则使用 SQL 逐条查询。

## 2.2 前端：交互界面

为了搭建一个尽可能轻量级的交互界面，本项目使用 Vue.js 作为前端框架，实现了一个无路由的单页面前端。

为了优化用户体验，前端除了搜索框、条件筛选器等必要元素，还纳入了如下交互元素：

- **无限下拉：**考虑到图片搜索引擎除了提供搜索结果的功能，还可以充当图片展厅，因此前端分页系统在设计上摒弃了传统的“跳转式”，改用“下拉式”，每当用户即将浏览完当前“页面”的图

<sup>1</sup>实际上，这一最大召回数量由 MAX\_RESULT 控制，默认为 500

片时，就会自动向后端发起搜索请求，获取更多图片，实现无缝衔接。

- **标签跳转：**标签是图片的一个重要属性，因此搜索入口除了手动输入标签，还支持点击搜索结果中的标签自动添加该标签作为筛选器并执行搜索。
- **相似查找：**以图搜图功能不仅仅支持上传本地图片，还可以直接点击结果卡片下方的“**搜索更多**”图标执行相似图片搜索。
- **隐藏式详细信息：**默认情况下，为了节约页面空间、减少对用户的干扰，只有图片本体和标签会被直接展示在页面上。不过如果用户想要获得更多关于此图片的信息，可以点击图片下方的“**下拉**”图标，查看文字描述等详细信息。

## 3 测试结果与分析

### 3.1 搜索功能测试

首先验证搜索功能的正确性与可靠性。

对于文本搜图，搜索引擎应当返回具有相关描述文本的图片。我们以关键词、短语两种类型的搜索语句为样例进行测试。下面列出每种搜索语句下前 5 条结果的文本描述。从中可以看出，排名靠前文档的的确为文本相似性较高的样本。

**关键词：**building people

1. In this picture we can see some **people** are standing and some **people** are walking on the pathway. Behind the **people** there is a fountain and a **building** with windows. In front of the **building** there is an architectural structure and poles with lights. Behind the **building** there is the sky.
2. In this image there are two **people** holding papers in their hands, in the background there are **people** standing and there is a **building** for that **building** there are doors and windows.
3. There are **people** standing, there is a **building** and a sky.
4. In a **building** there are three **people** walking on the floor and behind the **building** there are trees and behind the trees there is another **building**.
5. Some **people** wearing caps are sitting on bikes. On the second bike there is a flag. In the background there are some **people** standing. There is a **building** and trees. Also there are windows in the **building** in the background.

**短语：**children playing with toys

1. In this picture the big hall is there and so many people they are sitting on the **toys** and some of the **children** are **playing** the **toys** behind there is some trees are there the background is greenery.
2. In the image there are few kids standing on the floor **playing with toys** and behind them there is wall **with** sticky notes and charts all over it.
3. In this image I can see few **toys** are sitting on chairs.
4. This is an animated image in which there are **toys** and there are steps and there are lights on the wall.
5. In this picture, it seems to be there are toy cars and other **toys** in the image.

对于以图搜图，搜索引擎应当返回视觉上相似的图片。由于我们使用的是色彩信息，故结果图片应当在色彩上具有较高相似度。我们分别以沙滩（自然景观）、小鸟（动物）为样例进行测试，同样列出前 5 条搜索结果，见表 1。



表 1: 以图搜图结果，左侧为搜索用图，右侧为前 5 条结果图

从搜索结果来看，搜索引擎所返回的图片在视觉上（尤其是色彩搭配上）与搜索图基本一致，并且总能准确将相同的图片作为最相似图片返回。不过，受限于基于色块的图片特征提取算法，搜索引擎不能很好的识别出图片中的主体事物（如搜索鸟的图片会返回松鼠图片），想要解决这一缺陷需要引入基于纹理的特征提取。

### 3.2 搜索效率测试

一款好的搜索引擎应当在尽可能短的时间内向用户返回数据。目前数据库中图片数目为十万量级，而为了不让用户产生明显的延迟感，搜索耗时上限为百毫秒量级。不过，考虑到图片加载仍需要时间，搜索引擎本身的耗时还应当更小。

我们以搜索引擎后端计时和 Postman 请求耗时两个指标对搜索效率进行评价。为了覆盖更多的真实场景，我们设计了不同的搜索语句、筛选条件分别进行测试，结果如表 2 所示。

请求内容	筛选条件	后端计时 (毫秒)	Postman 计时 (毫秒)
building people	—	15.1	20
building people	tag=Man	23.1	28
building people	tag=Man&color=black	39.3	48
building people	pixels=s,m	18.2	25
building people	pixels=xl	46.1	50
children playing with toys	—	20.4	25
children playing with toys	tags=Boy,Clothing	39.0	44
—	tag=Food&color=orange	3.0	8
(local image)	—	172.0	180
(web image)	—	140.0	146

表 2: 不同场景下搜索耗时

从测试结果可知，基础的文本搜索耗时在十毫秒量级，完全满足要求；随着筛选条件的增多与条件命中率的降低，耗时仍会有所增加，不过基本不会超过 50 ms（表 2 中以 pixels=xl 为条件搜索 building people 时，已经无法命中任何结果）。当使用图片搜索时，耗时基本维持在 100 ms ~ 200 ms，亦属于可接受范围。

### 3.3 辅助功能与前端交互测试

在这部分测试中，我们以前端不同操作为例，验证前端交互逻辑与后端条件筛选功能。

首先测试标签筛选器，搜索“building people”，无筛选、筛选标签“Person”+“Building”两种情形下结果如图 2 所示。可以看到，后者搜索结果的标签中一定兼有“Person”和“Building”。

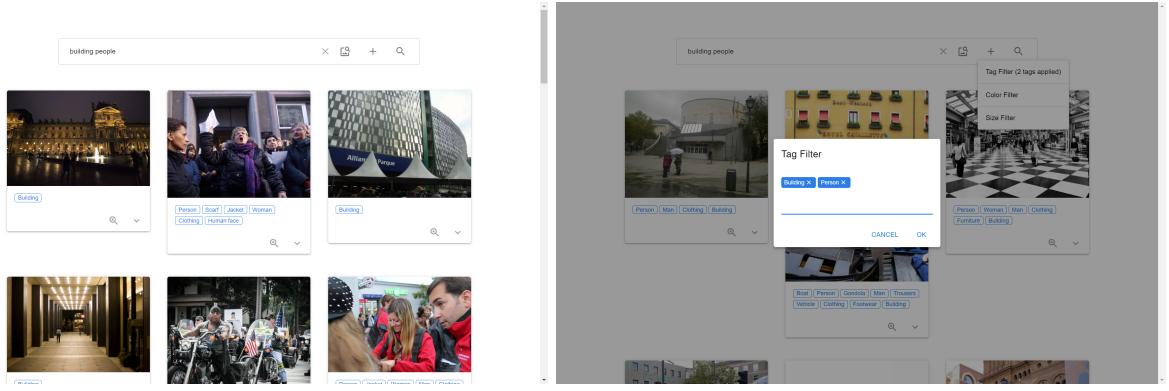


图 2: 标签筛选器测试 (左: 无筛选; 右: 筛选“Person” + “Building”标签)

然后测试颜色筛选器。搜索“flower”，无筛选、筛选颜色为“橙色”两种情形下结果如图 3 所示。可以看到，经过颜色筛选后，图片主色调均为橙色，也不再包含无筛选时返回的粉色花朵等。

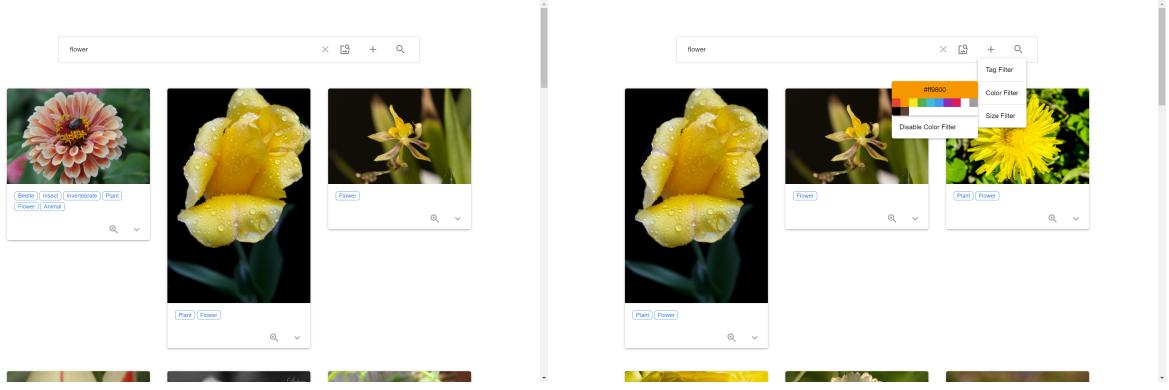


图 3: 颜色筛选器测试 (左: 无筛选; 右: 筛选“橙色”主导色)

下面测试尺寸筛选器。同样的搜索“flower”，筛选尺寸为“xs”、尺寸为“s”、尺寸为“m+”三种情形下结果如图 4 所示。可以看到，前两个搜索结果无交集，而第三次已经无法搜到满足要求的图片（因为该数据集中的图片尺寸普遍偏小）。如果一一查看图片的真实尺寸，可也证明其像素数大小关系是正确的。

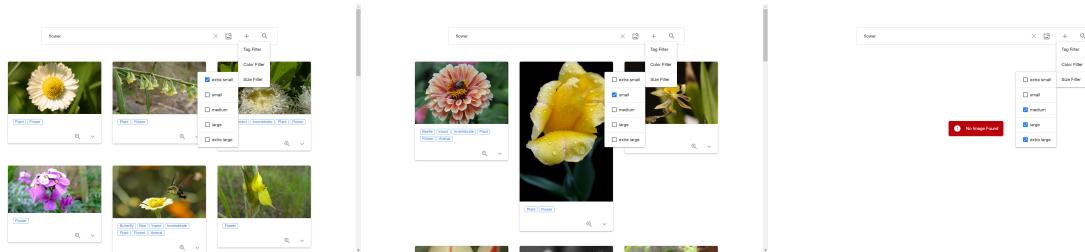


图 4: 尺寸筛选器测试 (左: 筛选“xs”尺寸; 中: 筛选“s”尺寸; 右: 筛选“m+”尺寸)

最后测试一些交互功能，包括无限下拉（分页系统）、标签跳转、搜索相似和隐藏式详细信息。相应截图在图 5、6、7、8 中给出，一切功能运作正常。

## 4 参考资料

本项目基于如下开源框架：

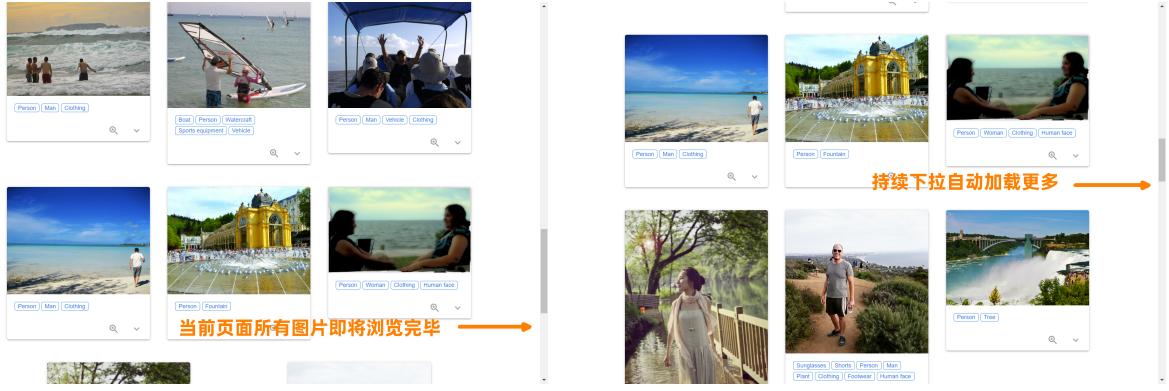


图 5: 无限下拉功能演示

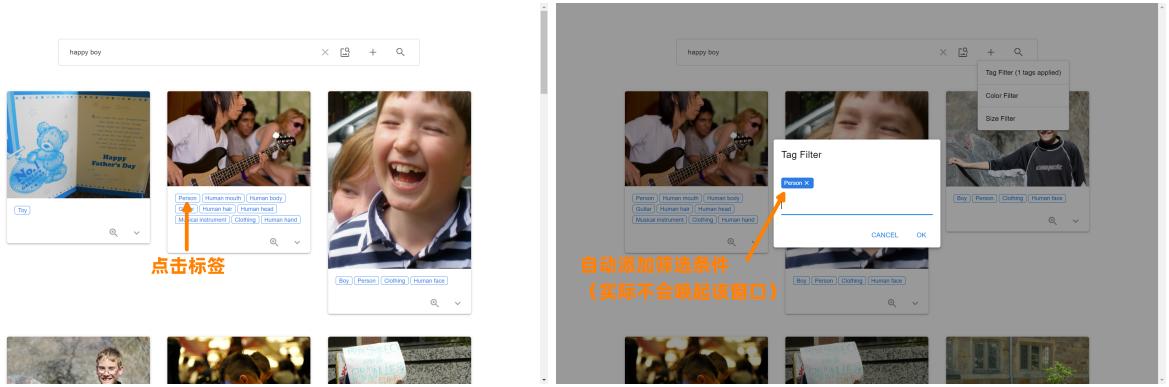


图 6: 标签跳转功能演示

- 前端: Vue.js
- UI: Quasar
- 后端网络服务: FastAPI

在一些核心代码编写上，参考了如下开源项目（在代码中相应位置也给出了 credit 与链接）：

- 在图片颜色直方图计算上，参考了 sherlockchou86 编写的 CBIR 图片搜索引擎
- 文本检索停用词表取自 sebleier 上传的停用词表

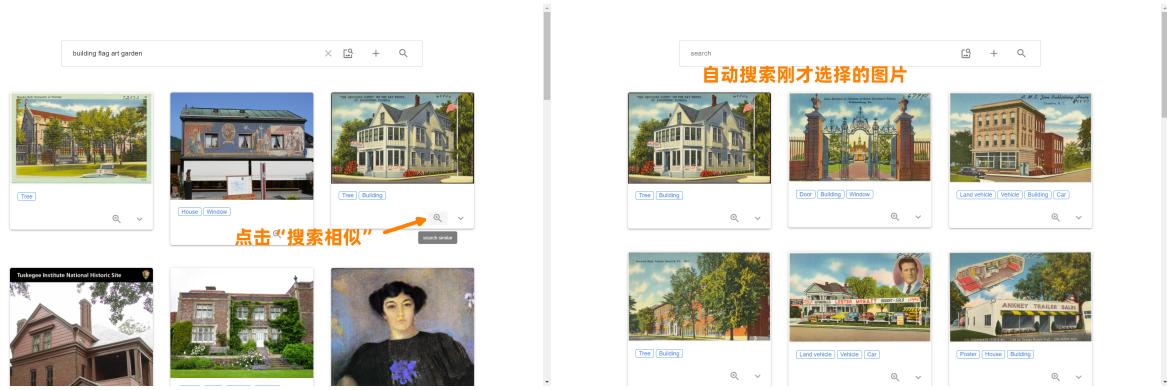


图 7: 搜索相似功能演示

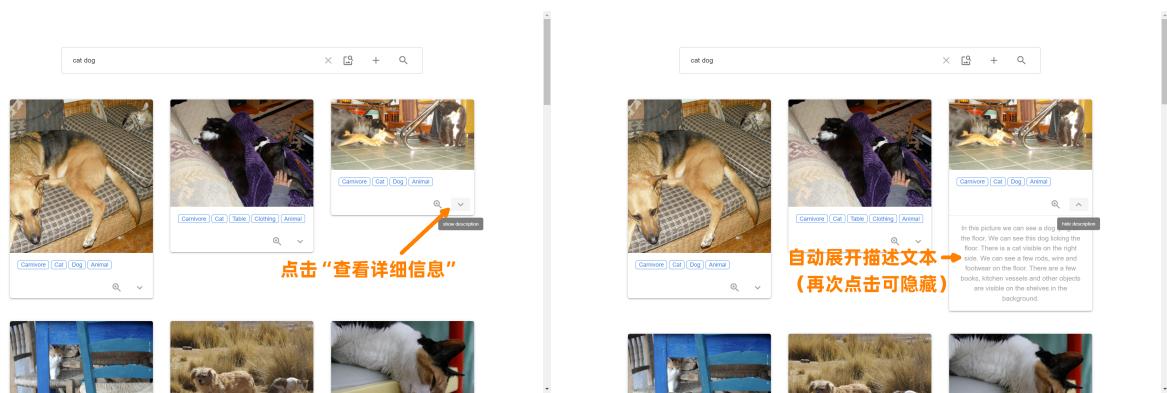


图 8: 隐藏式详细信息功能演示