

项目1：虚拟试穿体验

我们以动虚拟试穿为例，研究视觉大模型在时尚领域的应用

输入模特图与平铺图，商品的图像与模特的身体图像融合，来模拟穿衣后的效果。

静态试穿



现有多种开源的预训练的模型。

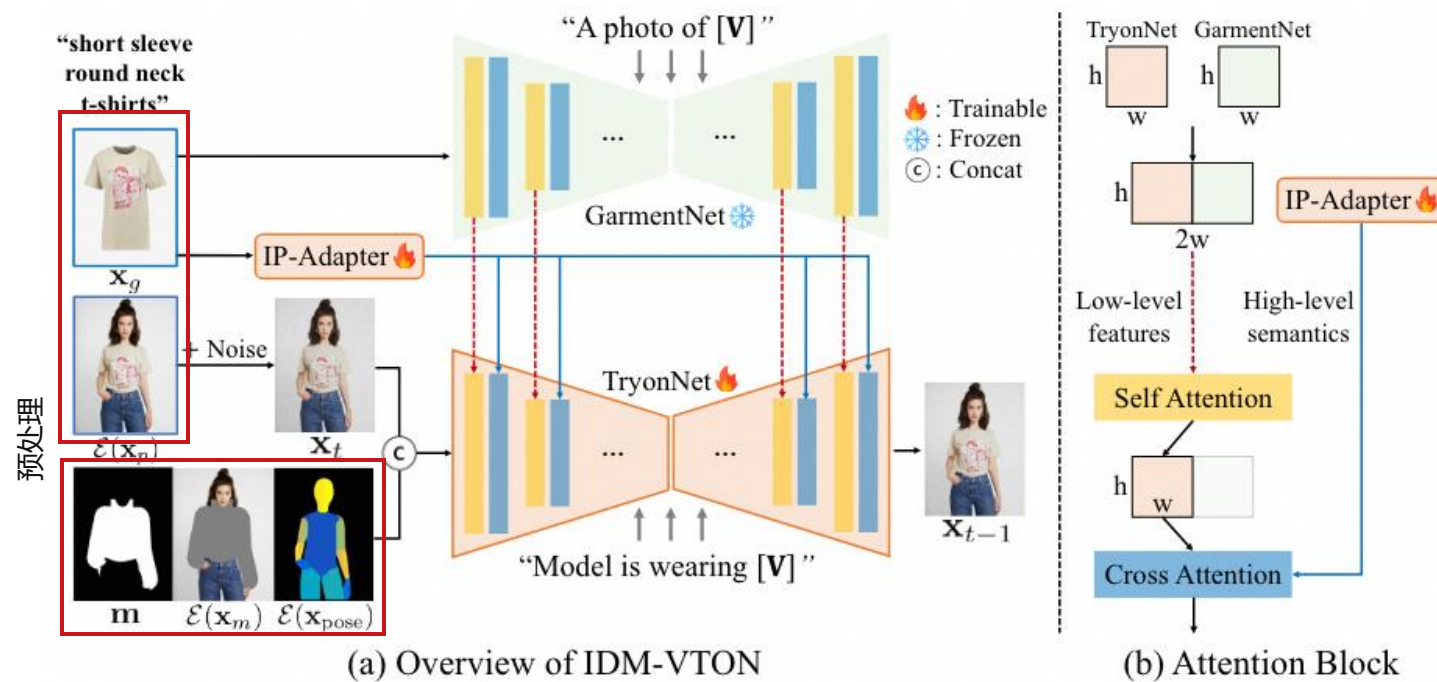
1. 但是由于公开数据集的数据bias (T恤居多)，在很多品类上表现不佳。
2. 现有模型由于数据集限制，通常只支持输入平铺服饰去给模特穿衣，不支持输入人身上的衣服换到模特上。即garment-person与person-person

在本项目中，我们需要收集这两方面的数据，并针对性的提升开源模型在这两方面的结果。



一阶段：静态试穿模型管线

以Stable Diffusion模型为骨架，采用双Unet架构，利用Reference Unet进行衣服特征的编码并注入人体之中。

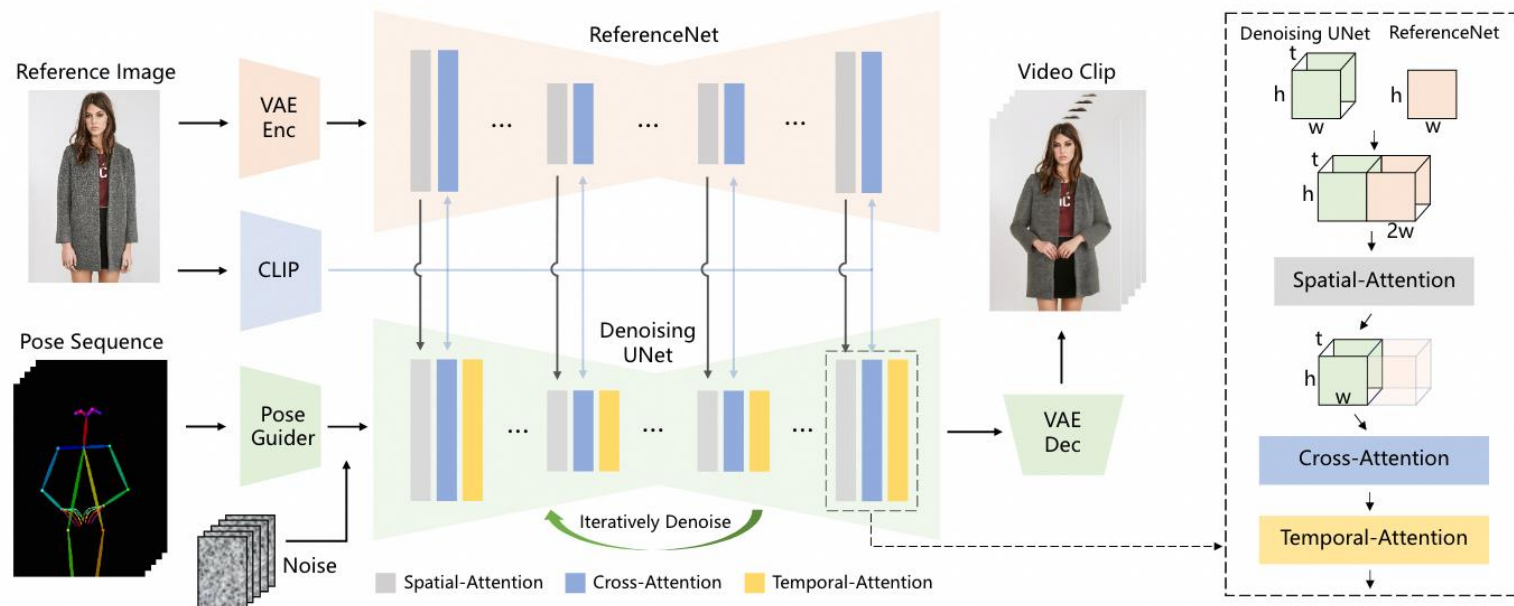


”

二阶段：动态驱动

动态驱动需要两种输入：参考图（即静态试穿结果图）以及人体骨架序列。
有预训练模型，实现一系列动作驱动。

同样采用双Unet架构，利用Reference Unet进行参考图的编码并注入人体之中。骨架序列通过轻量模型进行引导。





作业内容:



虚拟试衣的关键在于高质量数据的覆盖度，同学们使用高质量数据fine tune以后可以轻松达到超过baseline的效果。

具体内容:

1.【收集多品类的garment-person数据】每个小组分到一些品类，使用爬虫脚本，按照品类收集一批高质量的衣服对（平铺衣服以及模特上身图）。在组内去重后，至少人均1600条图片对，期望2000图片对。



2.【通过视频收集person-person数据】每个小组分到一些视频，利用视频分割工具，切出几秒至十几秒的clip，clip中只有一个占比适中的主体人物。并通过姿态检测模型，标注出一帧最标准的模特穿着衣服代表性的图片。在组内去重后，至少人均250条clip，期望300条clip。

3.【微调试衣模型】利用以上的garment-person数据和person-person数据，对现有sota模型进行训练（训练的模型的实验指标不作要求，因为同学训练资源不同。仅要求经过完整的训练过程）

4.【试衣结果动态化】完成一项最时尚的动态试衣！利用人体驱动模型对试衣结果进行驱动，形成时尚视频。

1-2-3-4项分别占比分数的25%-25%-20%-10%，另报告占比20%





动态虚拟试穿

- 算法推荐: OOTDiffusion, Animate Anyone: Consistent and Controllable Image-to-Video Synthesis for Character Animation
- 代码资源: <https://github.com/levihsu/OOTDiffusion>, <https://humanaigc.github.io/animate-anyone/>



加入组织



群聊：动态虚拟试穿作业群



该二维码7天内(9月25日前)有效，重新进入将更新

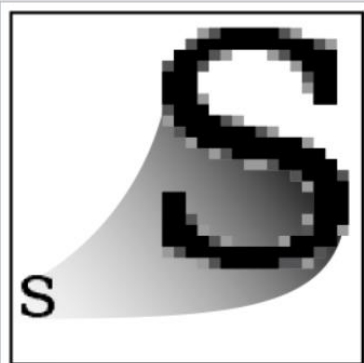


作业三：矢量图形生成



➤ 项目目标

- 了解矢量图形的意义以及大语言模型、多模态大模型在该领域的应用；
- 熟悉大语言模型训练、微调及部署全过程；
- 掌握通用计算机视觉项目开发基础技能；



Raster
GIF, JPEG, PNG

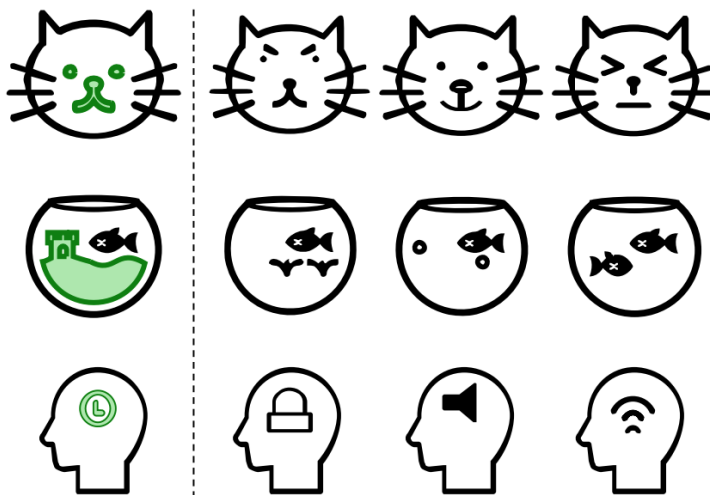


Vector
SVG

"cake, dessert,
sweet, food"



Random Editing



tiger and ...



smile and ...



cloud and ...

Text-Guided Editing



happy

sleep

one eye



folder



calendar



display



star



snow



rain

- Small File Size
- Scalability
- Easy to edit

```
<svg width="402pt" height="147pt" viewBox="0 0 401.50 147.00" xmlns="http://www.w3.org/2000/svg">
  <svg width="402pt" height="147pt" viewBox="0 0 401.50 147.00" xmlns="http://www.w3.org/2000/svg">
    <g id="graph0" class="graph" transform="scale(1 1) rotate(0) translate(4 143)">
      <polyline fill="white" stroke="transparent" points="-4,4 -4,-143 397.5,-143 397.5,4 -4,4" />
      <g id="node1" class="node">
        <polyline fill="white" stroke="black" points="137.5,-120.5 137.5,-120.5 237.5,-138.5 237.5,-120.5 137.5,-120.5" />
        <text text-anchor="middle" x="187.5" y="-126.5" font-family="Helvetica,sans-Serif" font-size="10.00">src/loss_Sort.h</text>
      </g>
    </g>
  </svg>
```



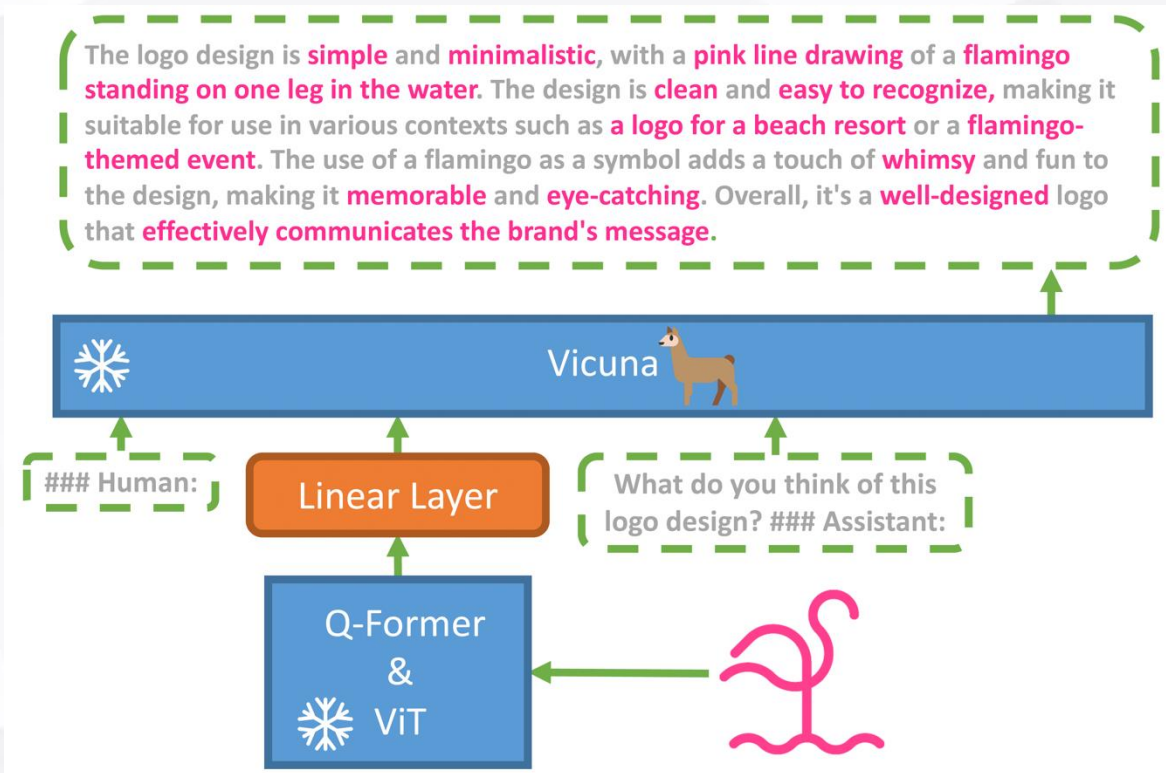


-



- 利用大模型对SVG进行标注;

将SVG光栅化成图像，并使用预训练模型为其生成文本描述



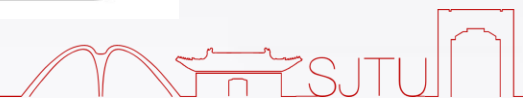
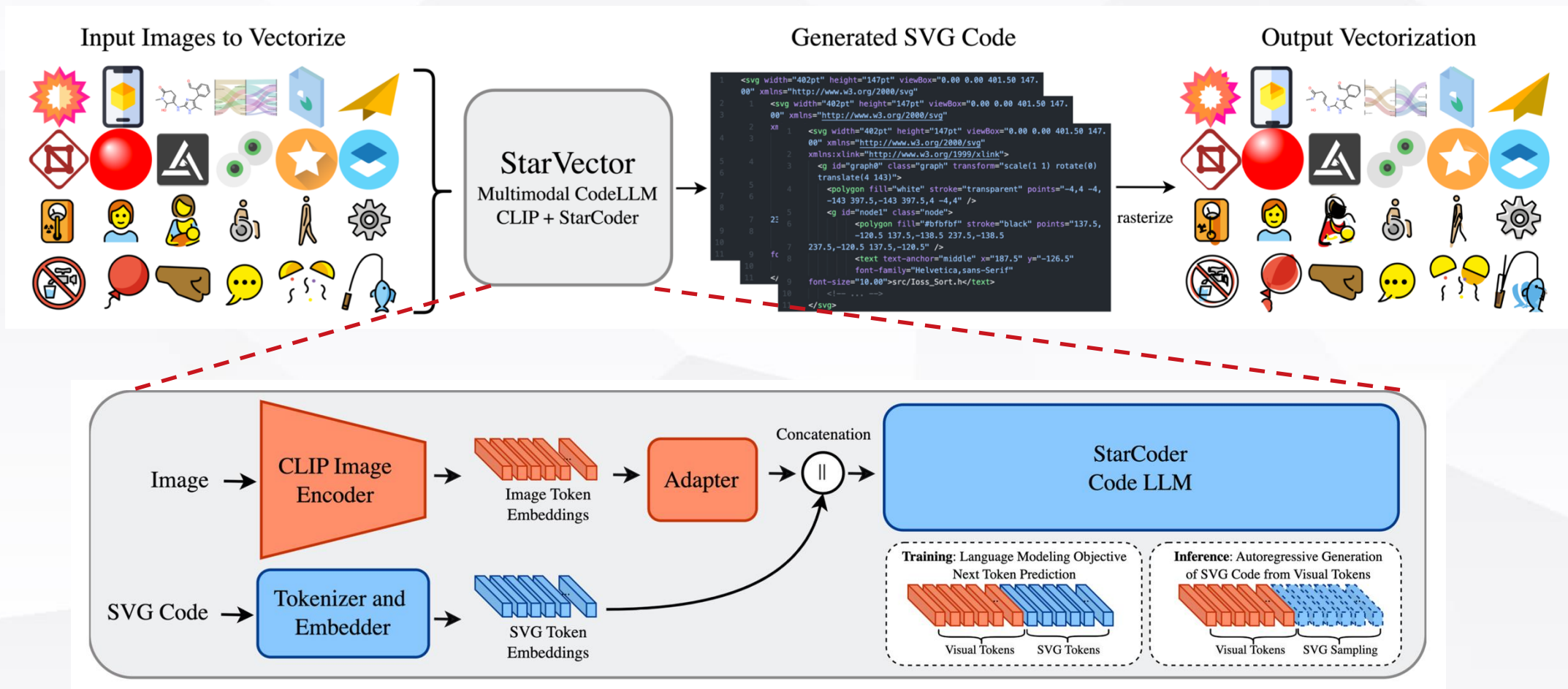
需设计量化指标对文本标注进行refine，修改或删除低质量标注



项目内容--阶段三



- 使用推荐的生成模型或自定义模型在收集的数据上部署训练;





作业内容



- 收集自己的矢量图及数据集
 - 利用爬虫（不强制）爬取数据，并对爬下来的数据进行清洗，删除**无明确标识意义**的数据；
- 利用大模型进行自动化标注，并**设计标注质量评价方案；提交对应代码；**
- 基于自己的数据集微调模型
 - 测试大模型 zero shot地生成SVG LOGO的能力；
 - 基于语言模型训练自己的SVG生成器；



📌 矢量图形生成

• 算法推荐:

- StarVector: Generating Scalable Vector Graphics Code from Images
- Vector Graphics Generation via Mutually Impulsed Dual-domain Diffusion
- Llama3.1 <https://llama.meta.com/>

• 工具推荐:

- ChatGPT 4o (可自行尝试其zero-shot SVG生成能力)



数据集具体要求

任务要求

1. 数据爬取：助教小组会提供爬虫示例脚本、示例网站、示例下载数据以及基于大模型的标注示例。示例脚本展示了一些常见的爬虫场景，请每个小组仔细学习后，了解爬虫基本思路之后再针对数据源网站自行撰写对应爬虫脚本，进行数据爬取。
2. 数据标注：爬取之后需要把人眼难以辨别含义的数据清除，并基于大模型添加英文描述。文本以json格式存储。
3. 设计基于大模型的方案评价标注质量，如验证标注文本与SVG数据匹配程度。

通过精心设计prompt，利用大模型生成SVG的描述文本

Meta:

SVG_02.parquet/less-than-2048/28905

Svg:



Query:

Review the image given, please give the text prompt that could generate a svg file whose rendered result is this image. Note that give the text prompt according to the following json format. ("Overall name": <answer>, "Overall event": <answer>, "Overall style": <answer>, "Overall colors": <answer>, "Overall shape": <answer>, "All parts": [{"Part item": <answer>, "Part style": <answer>, "Part colors": <answer>, "Part shape": <answer>, "Part size": <answer>, "Part position": <answer>, "Part relationship to entirety": <answer>, "Part extra description": <answer>}], "Overall extra description": <answer>}. Please note that there are some explanations below on how you should answer: 1. You should replace <answer> with your answer. If you don't have a suitable answer, you can answer "null". 2. For key "Overall name", please give the entirety a name in a few short words. 3. For key "Overall event", please briefly describe the event or action represented by the entirety in one sentence. 4. For key "Overall style", please answer overall style in a few short words. 5. For key "Overall colors", please answer in a few short words all the colors used for the entirety. 6. For key "Overall shape", if the entirety has clear semantic information, please answer "null" here. If the entirety does not have clear semantic information, please give a common shape noun to describe the shape of the entirety. Note that shape nouns are selected from the following list: Circle, Square, Rectangle, Triangle, Ellipse, Polygon, Parallelogram, Rhombus, Trapezoid, Star, Heart, Crescent, Diamond, Semicircle, Arrow. 7. For key "All parts", its value is a list of dictionaries containing detailed information about the various parts that make up the entirety. 8. For key "Part item" in one of values of key "All parts", please use a few short words to answer what this part represents, the thing, event or action. 9. For key "Part style" in one of values of key "All parts", please answer this part's style in a few short words. 10. For key "Part colors" in one of values of key "All parts", please answer in a few short words all the colors used for this part. 11. For key "Part shape" in one of values of key "All parts", please use a common shape noun to describe the shape of this part. Note that shape nouns are selected from the

Content Json:

```
{
  "Overall name": "Growth Chart"
  "Overall event": "Illustration of increasing data or growth"
  "Overall style": "Minimalistic"
  "Overall colors": "Yellow, Brown"
  "Overall shape": "Polygon"
  "All parts": [
    {
      "Part item": "Bars"
      "Part style": "Flat"
      "Part colors": "Yellow"
      "Part shape": "Rectangle"
      "Part size": "50%"
      "Part position": "0.2, 0.1, 0.8, 0.9"
      "Part relationship to entirety":
        "The bars represent the increasing data or growth over time."
      "Part extra description":
        "The bars are arranged in ascending order, indicating a positive trend."
    }
    {
      "Part item": "Arrow"
      "Part style": "Simplified"
      "Part colors": "Brown"
      "Part shape": "Arrow"
      "Part size": "10%"
      "Part position": "0.5, 0.5, 0.6, 0.6"
      "Part relationship to entirety":
        "The arrow indicates the direction of growth."
      "Part extra description":
        "The arrow points upwards, reinforcing the positive growth trend."
    }
  ]
  "Overall extra description":
    "The image is a simple, minimalist representation of growth or increase, using yellow bars and a brown arrow to convey the message. The bars are arranged in ascending order, and the arrow points upwards, emphasizing the positive trend."
}
```

数据要求

1. 两个小组独立收集，数据量要求：大于10000
2. 数据样本与标注文本成对存储
3. 将文件以下面形式压缩提交

组长姓名.zip

- 收集到的数据及标注
- 验证匹配的数据及标注



每个阶段提交结果，独立评分，互不影响

数据收集与清洗：十一月中旬提交数据与标注以及**评价方案**（评分占比**60%**）

模型训练与测试：十二月中旬提交代码与模型（评分占比20%）

项目报告：期末提交（评分占比20%）





大作业四：艺术图像编辑与生成

➤ 大作业目标

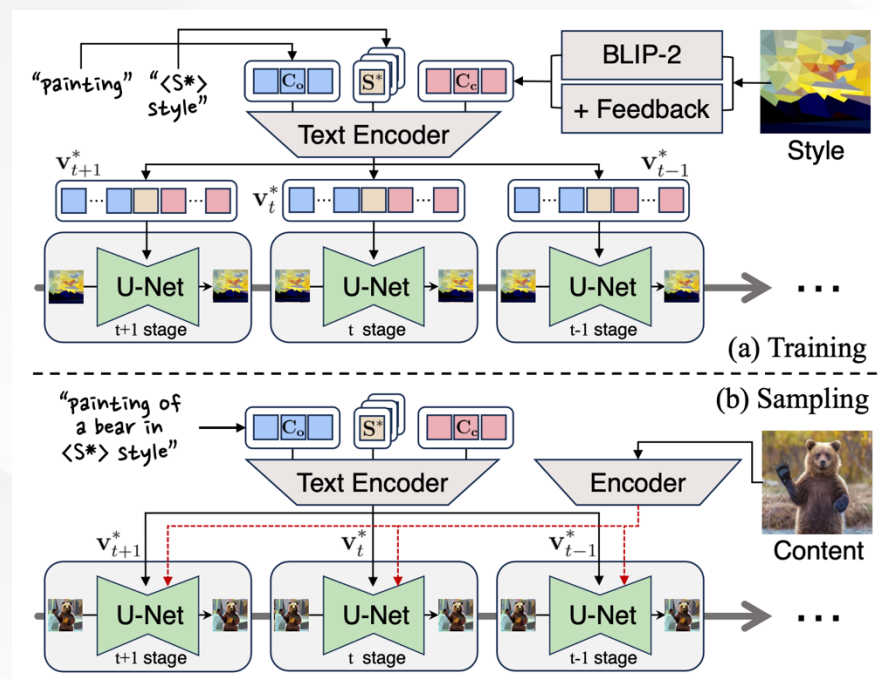
- 探究大模型对艺术风格图像的编辑与生成能力
- 收集艺术图像数据集，并基于此尝试大模型微调，提升大模型垂域能力



➤ 基于Textual Inversion的定制化Text-to-Image技术;

Textual Inversion:

通过学习特定文本提示的嵌入向量，使模型生成与这些嵌入向量相关的特定概念或风格的图像。





- 收集自己的艺术图像数据集
 - 利用爬虫（不强制）爬取数据，并对爬下来的数据进行清洗，删除低质量（如清晰度较低）的数据
 - 对数据进行归类整理
 - 共两个组别，其中一组为“插画-素描-（平面）设计”三类，另一组为“卡通（动漫）-油画-水墨画”三类
- 基于自己的数据集微调模型
 - 测试Stable Diffusion zero-shot地生成艺术图像的能力；
 - 基于收集的数据集，使用Textual Inversion技术来增强模型的艺术生成能力；



艺术图像生成

- 算法推荐:

- DreamStyler: Paint by Style Inversion with Text-to-Image Diffusion Models
- StableDiffusion: High-Resolution Image Synthesis with Latent Diffusion Models

- 开源代码推荐:

- <https://nmhkahn.github.io/dreamstyler/>
- <https://github.com/CompVis/latent-diffusion>



数据集具体要求

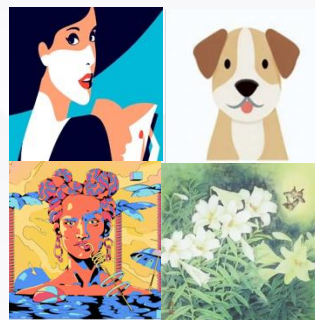
任务要求

1. 数据爬取：助教小组会提供爬虫**示例脚本**、**示例网站**与**示例下载数据**。示例脚本展示了一些常见的爬虫场景，请每个小组仔细学习后，了解爬虫基本思路之后再**针对数据源网站自行撰写对应爬虫脚本，进行数据爬取**。
2. 数据清洗：爬取之后需要把不符合类别、清晰度小于（400x400）的数据清除，并归类整理。

数据要求

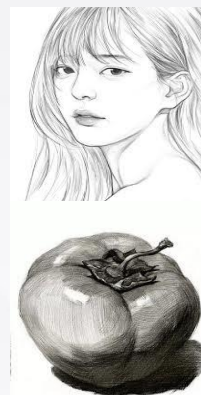
1. 两个小组分别收集“**插画-素描-（平面）设计**”三类以及“**卡通（动漫）-油画-水墨画**”三类艺术图像，每个类别的有效数据量要求：**10000张以上**
2. 将文件以下面形式压缩提交
 - 组长姓名.zip
 - |---插画
 - |---收集到的数据
 - |---清洗后的数据
 - |---素描
 -
3. **严禁使用生成式模型生成数据！**

➤ 插画

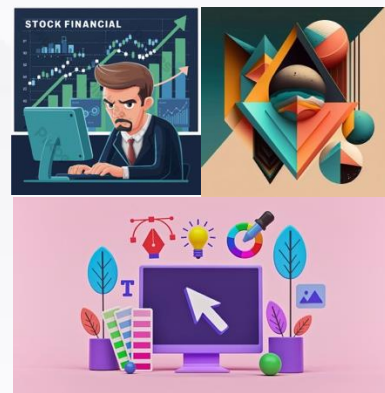


插画、版画、涂鸦
(大色块, 简约纹理)(人物、静物等)

➤ 素描



➤ 平面设计

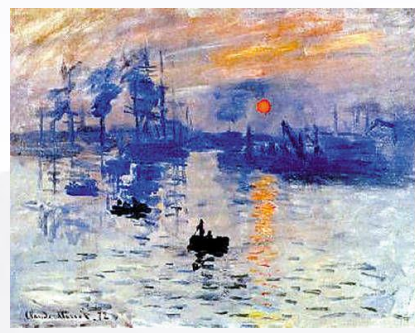


(海报、字体)

➤ 卡通



➤ 油画



(莫奈, 印象派)

➤ 水墨画





每个阶段提交结果，独立评分，互不影响

数据收集与清洗：十一月中旬提交数据（评分占比50%）

模型训练与测试：十二月中旬提交代码与模型（评分占比30%）

项目报告：期末提交（评分占比20%）





分组登记





加入组织



群聊：动态虚拟试穿作业群



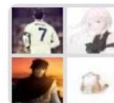
该二维码7天内(9月25日前)有效，重新进入将更新



群聊：三维物体生成



该二维码7天内(9月26日前)有效，重新进入将更新



群聊：矢量图形生成



该二维码7天内(9月26日前)有效，重新进入将更新



群聊：艺术图像生成



该二维码7天内(9月26日前)有效，重新进入将更新