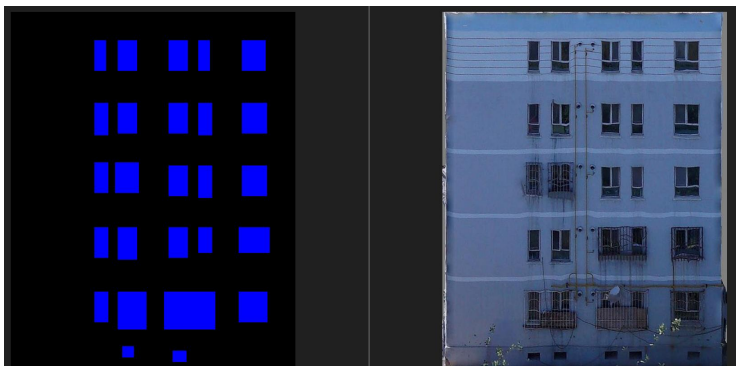
# 窗户无监督聚类实验报告

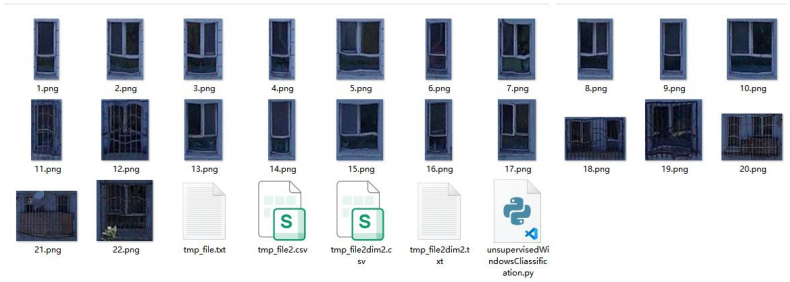
目录



## 数据

输入 22 张语义分割之后的窗户






## 无监督分类算法

具体算法：openai clip [openai/clip-vit-base-patch32 • Hugging Face]
编码结果：一张图像是 512 维的向量

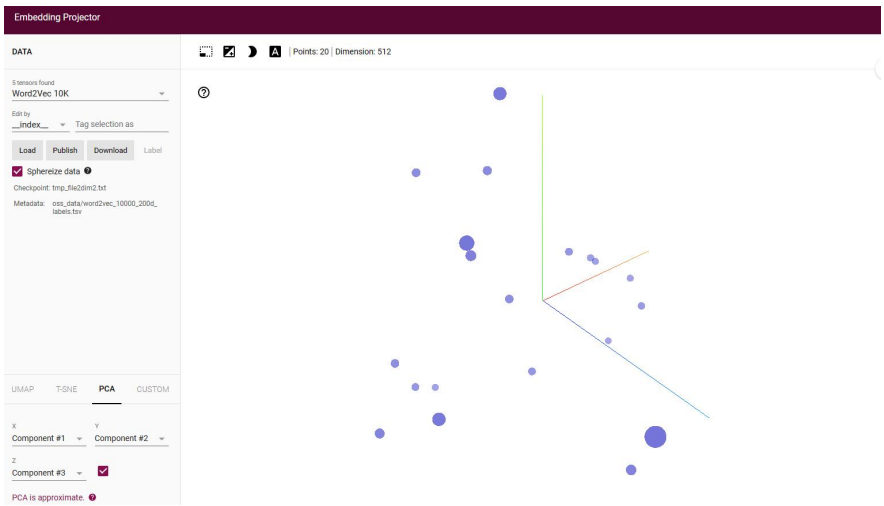
```python
from transformers import CLIPProcessor, CLIPModel
import torch

# device = "cuda"
model_id = "openai/clip-vit-base-patch32"

processor = CLIPProcessor.from_pretrained(model_id)
model = CLIPModel.from_pretrained(model_id)
# .to(device)

def embed_images(img_list):
    with torch.no_grad():
        images = processor(
            # text=None, images=img_list, return_tensors='pt')['pixel_values'].to(device)
            text=None, images=img_list, return_tensors='pt')['pixel_values']
        return model.get_image_features(images)

from PIL import Image

imgs = [Image.open('1.png'), Image.open('2.png'), Image.open('3.png'), Image.open('4.png'), Image.open('5.png'),
Image.open('6.png'), Image.open('7.png'), Image.open('8.png'), Image.open('9.png'), Image.open('10.png'),
Image.open('11.png'), Image.open('12.png'), Image.open('13.png'), Image.open('14.png'), Image.open('15.png'),
Image.open('16.png'), Image.open('17.png'), Image.open('18.png'), Image.open('19.png'), Image.open('20.png')]

all_emb = embed_images(imgs)
all_emb = all_emb.numpy()
print(all_emb[0])
print(len(all_emb[0]))
# print(embedded)

with open('tmp_file2dim2.txt', 'w') as f:
    csv.writer(f, delimiter='\t').writerows(all_emb)
```

```
(modelscope) PS C:\Users\yiper\Desktop\20230908_142329_test\windows> python .\unsupervisedWindowsClassification.py
C:\Users\yiper\.conda\envs\modelscope\lib\site-packages\requests\__init__.py:114: RequestsDependencyWarning: urllib
  RequestsDependencyWarning,
[ 5.07458448e-01  5.72981983e-02 -8.63567108e-01 -2.23102301e-01
  3.05416226e-01 -1.91708134e-01  5.66733927e-02  3.40316355e-01
  3.13883781e-01  6.81842148e-01  7.20947981e-02  3.12630534e-01
  7.52285063e-01  1.66715324e-01 -2.75900866e-01  2.01843209e-01
  2.16172948e-01 -4.59776521e-02  1.91990692e-01 -5.21177888e-01
  5.82653539e-01 -9.75666866e-02  1.27286698e-01  2.10811438e-01
  4.27322537e-02 -8.96735191e-02  1.14492866e-01 -3.37717772e-01
  4.73675877e-02  1.83211781e-02 -2.46113539e-02 -2.96485603e-01
 -1.58143528e-01 -2.17324868e-02 -1.25937343e-01 -4.86688633e-01
 -1.81319514e-01 -2.02645898e-01 -8.46877813e-01  8.90737697e-02
 -8.76127148e-01 -4.71927817e-01 -3.30566188e-01 -2.76625514e-01
 -9.14801218e-02  1.10986613e+00  1.31502087e-01  3.23626518e-01
 -3.25269252e-03 -4.74299312e-01  3.21864814e-01  2.33597189e-01
  2.50973701e-01 -1.86641440e-01 -3.53211731e-01 -1.26434386e-01
 -1.63507015e-01 -1.84681201e-01 -1.95235431e-01 -4.20335472e-01
  6.63458122e-01 -1.57432850e-01 -2.23836900e-01  3.41991857e-02
 -5.97235113e-02  2.62022913e-01 -2.32686877e-01 -1.00811028e+00
 -3.59447428e-01  6.45756871e-02 -2.73209751e-01 -3.78513962e-01
 -7.39168938e-02  1.52921349e-01 -4.27753866e-01  5.68813205e-01
  2.26582687e-01 -5.59353769e-01  4.25226390e-02 -1.07482774e-02
  2.77952850e-01  8.46222714e-02  4.34485148e-01 -9.44138169e-01
  6.82828277e-01  9.68528688e-02  8.97156000e-01  2.44786248e-01
 -4.12602544e-01 -3.58172859e-02  6.39681573e-02  3.12230326e-02
 -7.44786249e+00  3.72129887e-01  1.16776623e-01 -1.46287675e-01
  1.15508288e-01  7.31541887e-02  5.21937132e-01  5.93448430e-02
  4.39355597e-02 -2.57486741e-01  3.49766947e-02 -2.24270821e-01
  2.77662131e-01  1.55472249e-01 -1.07932842e+00  3.84187579e-01
  2.82369316e-01 -5.09787798e-02  2.15641489e-01 -1.55624747e-02
 -9.98347117e-02 -8.13783109e-02  2.81252503e-01  3.56535635e-01
 -5.55791855e-02 -6.85856256e-03 -9.72805505e-02  6.02611880e-01
 -4.77950722e-02 -1.15447134e-01  2.43620366e-01  3.33985239e-01
  2.04624087e-02  1.88416070e-02  1.98810534e-01  6.75924271e-02
  4.96282583e-02  2.93925468e-01 -6.47866161e-02 -1.12654179e-01
 -2.58082935e-01  9.15809701e-01  2.62866043e-02  4.50438632e-01
  2.22388297e-01 -4.87392753e-01 -3.61195326e-01  2.10699484e-01
  1.89006239e-01  1.92773670e-01 -3.03633153e-01  1.76501453e-01
 -2.73723245e-01 -1.20933518e-01  2.11575851e-02  4.58032881e-01
  2.76247283e-01 -2.92185913e-01  1.89375192e-01 -6.25811995e-02
  3.43173563e-01  1.85995827e-02 -2.77748784e-01 -6.15134776e-01
  3.52687985e-01  1.26968130e-01 -2.27311820e-01 -1.50031090e-01
  1.25668885e-01  2.76848823e-02 -7.32178986e-02 -1.63957998e-01
  9.82996821e-02  5.78589737e-01 -9.95401517e-02 -2.11912766e-02
```

# 实验结论

不可用，此方法不行

512 维的图像向量 PCA 降为至 3 维的效果如下，并没有出现聚类，应该是分成 3 或 4 类



# 解决方法

1. 使用其他深度学习算法实验
2. 使用一些照片进行微调，再用 clip 算法编码
3. 使用其他传统的特征提取算法实验