

- [VisQ-Search-Engine](#)
- [Paper](#)
  - [Project description](#)
  - [How to use](#)
    - [Installation](#)
    - [Data preparation](#)
    - [Train](#)
    - [Query](#)
  - [Acknowledgements](#)
  - [License](#)

# VisQ-Search-Engine

---

A Visual Query-Driven Search Engine for Brain Tissue Image Analysis

## Paper

---



## Project description

---

- will update later...

## How to use

---

## Installation

You can use `python setup.py develop` to install the environment directly, or manually install the following packages. `requirement pkgs=[ 'numpy', 'torch=2.3.0',`

'torchvision', 'six', 'h5py', 'Pillow', 'scipy', 'scikit-learn', 'metric-learn', 'faiss\_gpu']

- `cd ClusterContrast`
- `pip install -e .`

## Data preparation

You need to train the whole dataset before using the query search engine, after it has been trained, the user can use the list of the locations as the query engine input. Here is the method for preparing the dataset:

```
python make_blindDS_mauai.py
--INPUT_DIR <Path to the input dir containing biomarker images>
--OUTPUT_DIR <Path to the output dir>
--BBXS_FILE <Path to the bbxs_detection.txt file generated from cell
nuclei detectio module>
--channel_names <List of filenames for channels in the order: [dapi,
histone, neun, s100, olig2, iba1, reca1]>
```

Alternatively, there are several default variables that you can change by your need, please check the code in the file. The input biomarker images are the whole brain images and the ouput are the cropped [175,172,10] patches. Below is a sample of how to run the code.

```
python make_blindDS_mauai.py \ --
INPUT_DIR=/project/roysam/rwmills/data/brain/MDA_GBM/1168457/intra_cor
rected/ \ --OUTPUT_DIR=/project/roysam/rwmills/repos/cluster-contrast-
reid/examples/data/MDA_GBM_1168457_whole.2/ \ --
BBXS_FILE=/project/roysam/rwmills/data/brain/MDA_GBM/1168457/detection
_results/bbxs_detection.txt \ --DAPI=R1C1.tif \ --HISTONES=R1C2.tif \
--NEUN=R1C3.tif \ --S100=R1C4.tif \ --OLIG2=R1C5.tif \ --IBA1=R1C6.tif
\ --RECA1=R2C2.tif \ --other1=R2C3.tif \ --other2=R2C4.tif \ --
other3=R2C5.tif \
```

You can see the example of the data in the bbs\_detection.txt. We recommand you to set the file arc as below:

```
cluster-contrast-reid
├─ clustercontrast
├─ exaples
└─ data
```

```
| └─ logs
| └─ pretrained
├─ results
├─ runs
├─ bash.sh
└─ setup.py
```

## Train

To train the network, we need several args, here is the explanation:

```
CUDA_VISIBLE_DEVICES=0,1,2,3 In default, we train the network in 4-
GPUs, it corresponding the variavle -j, if you are using other nubmer
of GPUs, you need to change the variable -j to the number of the Gpus
-b batch size
-a backbone network
--iters number of the epoch
--momentum the momentum of the encoder update rate
-- eps max neighbor distance for DBSCAN
-- k1 hyperparameter for KNN
-- k2 hyperparameter for outline
```

One example for training:

```
CUDA_VISIBLE_DEVICES=0,1,2,3 python
examples/cluster_contrast_train_usl_infomap.py -b 256 -a resnet50 -d
market1501 --iters 200 --momentum 0.1 --eps 0.5 --k1 15 --k2 4 --num-
instances 16 --logs-dir /project/roysam/rwmills/repos/cluster-
contrast-reid/examples/logs/infomap/ --height 50 --width 50
```

## Query

Run the code in BooleanUser\_lhuang.ipynb to see how to interact with the machine learning model with your query and the output display. Before that, you need to prepare the following files:

- pretrained matrix
- atlas\_borders
- Target whole brain image

...More Details...

# Acknowledgements

---

- Will update later

# License

---

If you use this codebase in your work, please consider citing:

```
@article{update later,  
  author = {update later},  
  title = {update later},  
  journal = {update later},  
  year = {update later},  
  doi = {update later},  
}
```