RetCCL: Clustering-guided contrastive learning for whole-slide image retrieval

Xiyue Wang , Yuexi Du, Sen Yang, Jun Zhang, Minghui Wang,
Jing Zhang, Wei Yang, Junzhou Huang, Xiao Han

College of Biomedical Engineering, Sichuan University, Chengdu, China
College of Computer Science, Sichuan University, Chengdu, China
College of Engineering, University of Michigan, Ann Arbor, United States
Tencent AI Lab, Shenzhen, China

Medical Image Analysis 2023

Report: Yu-Chen Lai

Data: 2023.05.11.

Outline

- Introduction
- Method
  1. CCL-based feature extractor
  2. WSI retrieval process
- Experimental results
- Conclusion

1. Introduction

- **WSI retrieval** has recently attracted growing, which can return a series of similar WSIs from a historically characterized database when given a WSI for a query.
- These retrieved WSIs with associated diagnosis information can help provide high interpretability, making it possible in **clinical diagnosis,** medical research, and trainee education.

- **Challeng**
  1. **Effective feature extraction** is very challenging due to the enormous heterogeneity within WSIs and intra-/inter-class variations across WSIs.
  2. It is more desirable to find WSIs in which there exist **diagnosis-relevant** regions/patches rather than retrieving WSIs with **global similarity**.

1. Introduction

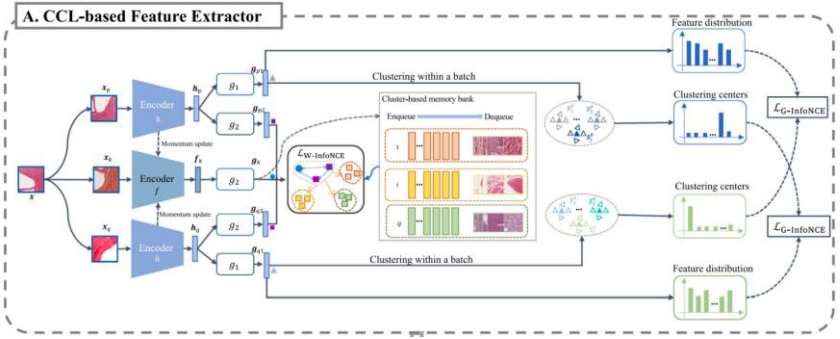
- Yottixel [**1**] and FISH [**2**] depend entirely or partly on the **ImageNet data**, which may result in suboptimal performance due to the domain difference between natural and pathological images.
- For histopathological images, **negative pairs** in the contrastive learning setting may be composed of **highly related samples**, which could **confuse** the network training process.

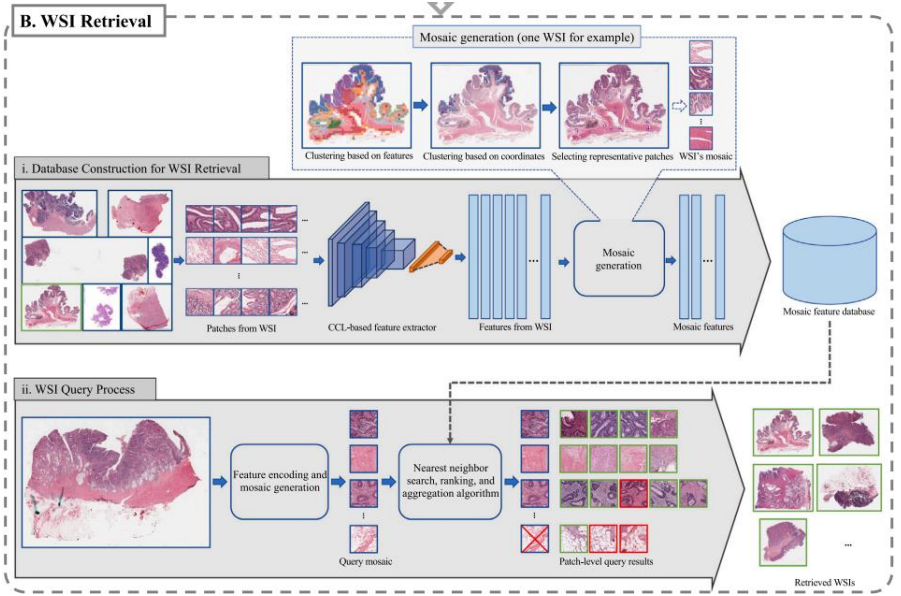
- **<u>Goal</u>**
  1. Robust content **feature extraction**
  2. A global **aggregation approach** on the local patch retrieval results to find the **most similar WSIs**.

*[**1**] Shivam Kalra, et al. "Yottixel–an image search engine for large archives of histopathology whole slide images." Medical Image Analysis. 2020.*
*[**2**] Chengkuan Chen, et al. "Fast and scalable search of whole-slide images via self-supervised deep learning." Nature Biomedical Engineering. 2022.*

2. Methods

- The overview of WSI retrieval framework (**RetCCL**) is implemented using a two-stage strategy, including the **CCL-based feature extractor** and the **WSI retrieval process**.





2. Preliminary of contrastive learning

- Given an image $\boldsymbol{x}$ and its two different augmented views: $\boldsymbol{x}_q$ and $\boldsymbol{x}_k$,
- Pull closer the features ($\boldsymbol{q}$ and $\boldsymbol{k}$+) of views from the same image.
- Repel away the features ($\boldsymbol{q}$ and $\boldsymbol{k}$−) of views from different images.

$$\mathcal{L}_{\text{InfoNCE}} = -\log \frac{\exp(\boldsymbol{q} \cdot \boldsymbol{k}^+/\tau)}{\exp(\boldsymbol{q} \cdot \boldsymbol{k}^+/\tau) + \sum_{i=1}^{L} \exp(\boldsymbol{q} \cdot \boldsymbol{k}_i^-/\tau)}$$

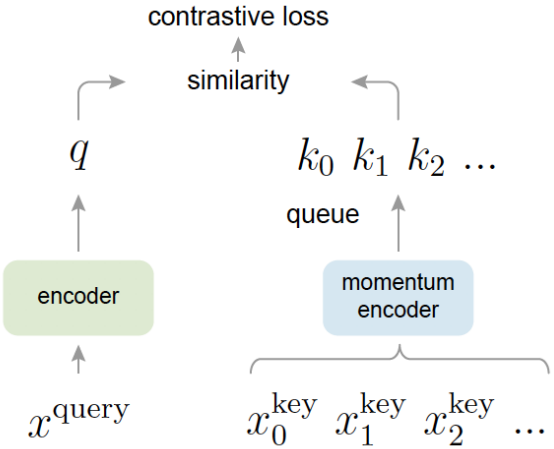


Figure 1. Moco [3]

[3] Kaiming He, et al. "Momentum contrast for unsupervised visual representation learning." CVPR. 2020.

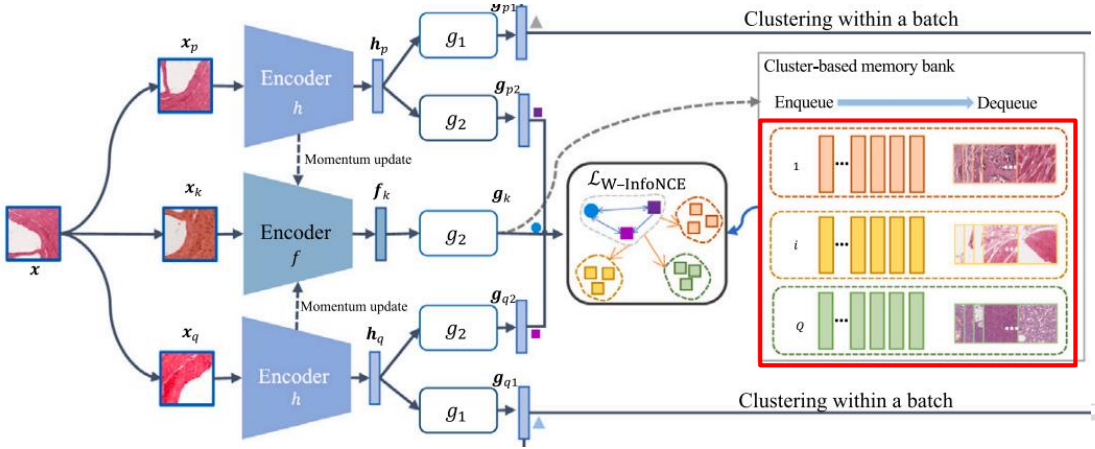
2. Clustering-guided contrastive learning

- **Problem**:
  - The basic assumption about the negative samples is not suitable for WSI.
  - Just like [4] have claimed, there may exist possible **highly correlated samples**, which **should be considered as positive samples** with respect to the anchor sample x_q, while they are repelled from the anchor sample in the setting of the standard contrastive learning.
- **Solution**:
  - **Weighted InfoNCE** ($\mathcal{L}_{\text{W-InfoNCE}}$)
    - Based on a subqueue strategy to reduce the effect of possible false-negative samples in contrastive learning.
  - **Group-level InfoNCE** ($\mathcal{L}_{\text{G-InfoNCE}}$) from [4]
    - Encourages the anchor sample and its nearest group center to have higher similarity while enforcing the anchor sample and the remaining group centers to have a lower similarity.

[4] Xudong Wang, Ziwei Liu, and Stella X. Yu. "Unsupervised feature learning by cross-level instance-group discrimination." CVPR. 2021.

2.1. Online Clustering-guided Memory Bank Construction

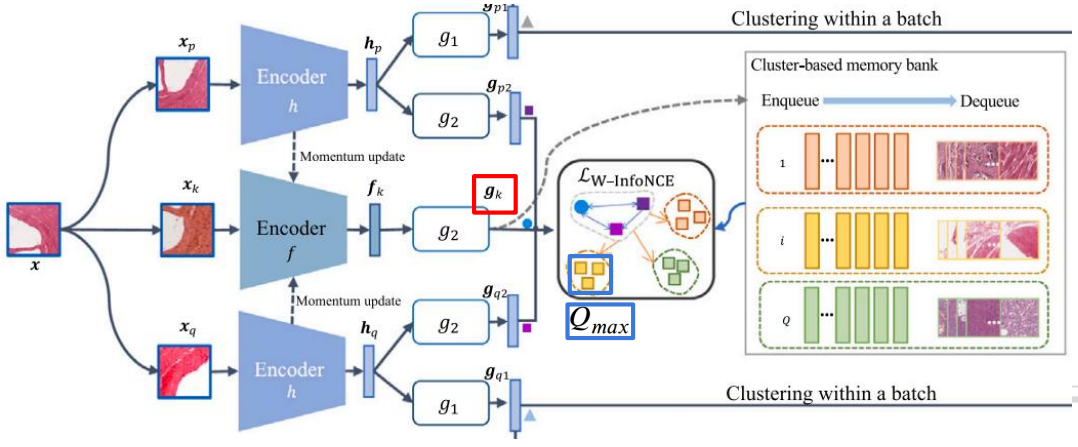
- **Goal**: Reduce the influence of potential false-negative samples.
  - A **weighted InfoNCE** loss is proposed to give less weight on these false-negative-like samples with respect to the anchor feature embedding.
- **Step1.**
  - All negative samples within the memory bank are first clustered into Q classes using the K-means approach, which are called Q sub-memory queues.
  - Their centroids are represented as $\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_j, \ldots, \boldsymbol{c}_Q\}$.



2.1. Online Clustering-guided Memory Bank Construction

- **Step 2.**
  - The **similarity scores** between the input feature g_k and each centroid $c_j (j = 1, 2, \dots, Q)$ are calculated as $\{Sim_1, \dots, Sim_j, \dots, Sim_Q\}$.
  - The **maximum** of these similarity scores can be obtained as Sim_{max}, which corresponds to the cluster Q_{max} whose centroid is most similar to g_k.
  - The weight $\phi(g_{k_i^-})$ for each negative sample in the memory bank can be calculated by

$$\phi(g_{k_i^-}) = \begin{cases} w, \text{ if } g_{k_i^-} \in Q_{max} \\ 1, \text{ otherwise} \end{cases}, \text{ where } w \in [0, 1)$$



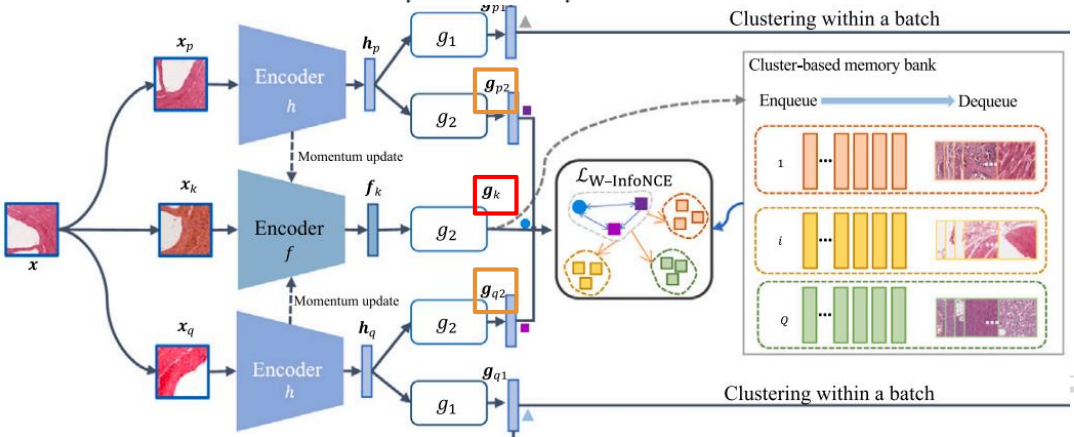
2.1. Online Clustering-guided Memory Bank Construction

- **Step 3.**
  - The **weighted InfoNCE loss** W-InfoNCE can be defined as

$$\mathcal{L}_{\text{W-InfoNCE}} =$$

$$-\frac{1}{2}\log\frac{\exp(\boldsymbol{g}_{p_2}\cdot\boldsymbol{g}_k/\tau)}{\exp(\boldsymbol{g}_{p_2}\cdot\boldsymbol{g}_k/\tau)+\sum_{i=1}^{L}\exp(\phi(\boldsymbol{g}_{k_i^-})\cdot\boldsymbol{g}_{p_2}\cdot\boldsymbol{g}_{k_i^-}/\tau)}$$

$$-\frac{1}{2}\log\frac{\exp(\boldsymbol{g}_{q_2}\cdot\boldsymbol{g}_k/\tau)}{\exp(\boldsymbol{g}_{q_2}\cdot\boldsymbol{g}_k/\tau)+\sum_{i=1}^{L}\exp(\phi(\boldsymbol{g}_{k_i^-})\cdot\boldsymbol{g}_{q_2}\cdot\boldsymbol{g}_{k_i^-}/\tau)}$$



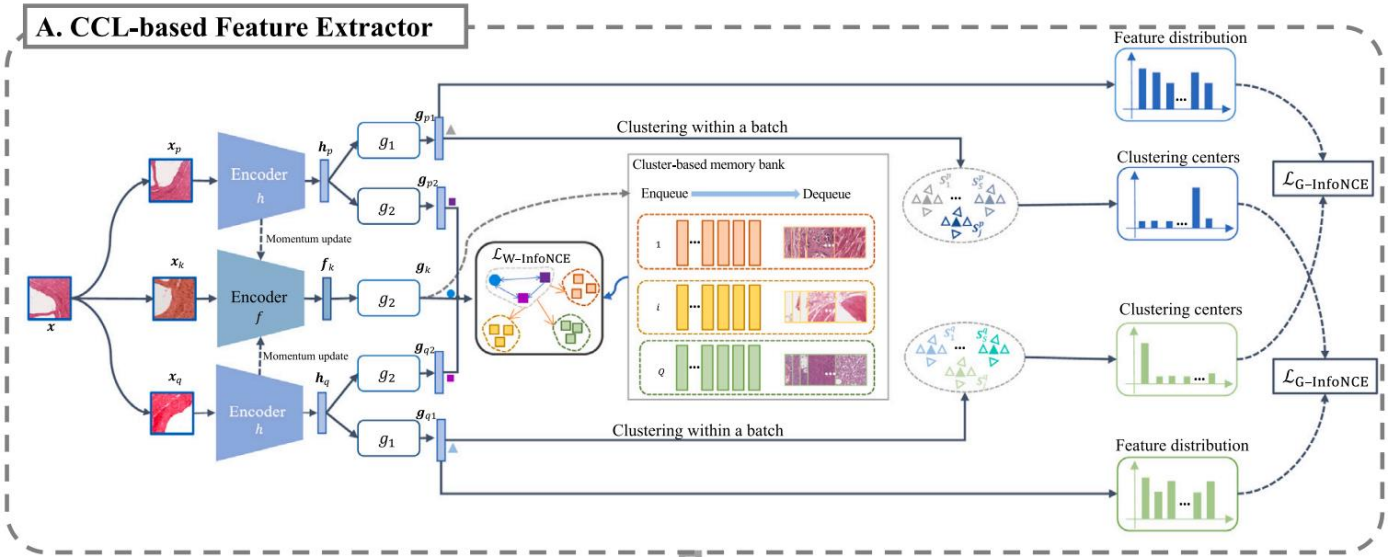
2.1. Online Clustering-guided Memory Bank Construction

- **Clustering Process.**
  - At the **beginning**
    - Initialize the Q centroids of these sub-memory queues, we randomly feed T histopathological images into our encoder that is **initialized** by its pretrained weights on the **ImageNet** data.
  - During each **iteration**
    - Each cluster centroid is updated by $\boldsymbol{c}_{j*} \leftarrow m_c \boldsymbol{c}_j + (1 - m_c) \cdot \frac{1}{\left|\mathcal{B}_j\right|} \sum_{\boldsymbol{g}_k^i \in \mathcal{B}_j} \boldsymbol{g}_k^i$

      where $m_c \in [0, 1]$ represents a weighting factor, $\mathcal{B}_j$ denotes the feature set of the jth class(cluster) in the current mini-batch, $\boldsymbol{g}_k^i$ represents the ith feature vector in the mini-batch
  - At each **epoch**
    - All clustering centroids will be updated by re-clustering all negative samples in the memory bank.

2.2. Group-level Discrimination

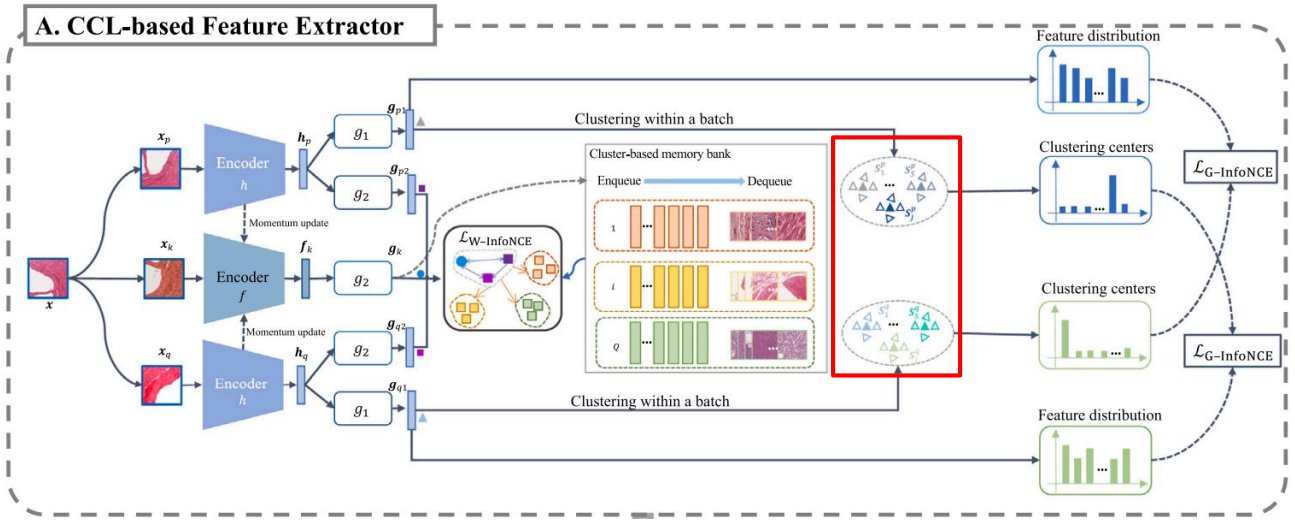
- **Goal:** further mitigate the unbalanced positive/negative sample ratio
  - Add the **cross-level discrimination (CLD)** [4] as an auxiliary branch.



[4] Xudong Wang, Ziwei Liu, and Stella X. Yu. "Unsupervised feature learning by cross-level instance-group discrimination." CVPR. 2021.

2.2. Group-level Discrimination

- **Step 1.**
  - Embeddings from all samples in one **mini-batch** are then clustered into S clusters for each of the two augmented view branches, and their centroids are denoted respectively as S_j^p and S_j^q, where $j \in [1, 2, \ldots, S]$.



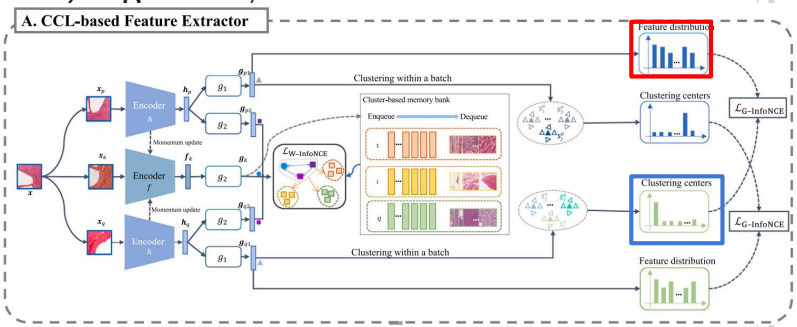
2.2. Group-level Discrimination

- **Step 2.**
  - Given a query instance, we compare its augmented view embedding ($\boldsymbol{g}_{q_1}$ or $\boldsymbol{g}_{p_1}$) with cluster centroids in the opposite branch to define positive and negative pairs for computing the **group-level InfoNCE loss**.
  - For example, given $\boldsymbol{g}_{q_1}$
    - $\boldsymbol{S}^{q+}$: **positive** sample, the closest centroid in the x_q branch.
    - $\boldsymbol{S}_i^{q-}$: **negative** samples, the remaining S**-1** centroids, where $i \in [1, 2, \ldots, S-1]$.
  - The **group-level InfoNCE loss** $\mathcal{L}_{\text{G-InfoNCE}}$ (CLD loss) is given by

$$\mathcal{L}_{\text{G-InfoNCE}} =$$

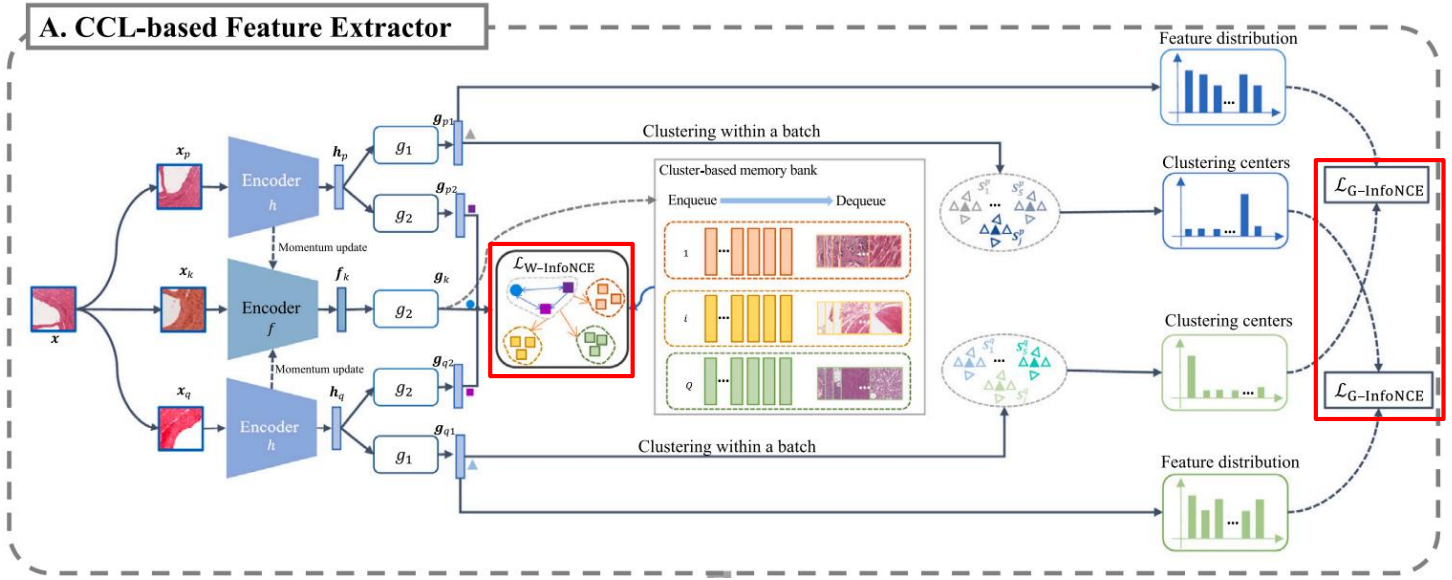
$$-\frac{1}{2}\log\frac{\exp(\boldsymbol{g}_{p_1}\cdot\boldsymbol{S}^{q+}/\tau)}{\exp(\boldsymbol{g}_{p_1}\cdot\boldsymbol{S}^{q+}/\tau)+\sum_{i=1}^{S-1}\exp(\boldsymbol{g}_{p_1}\cdot\boldsymbol{S}_i^{q-}/\tau)}$$

$$-\frac{1}{2}\log\frac{\exp(\boldsymbol{g}_{q_1}\cdot\boldsymbol{S}^{p+}/\tau)}{\exp(\boldsymbol{g}_{q_1}\cdot\boldsymbol{S}^{p+}/\tau)+\sum_{i=1}^{S-1}\exp(\boldsymbol{g}_{q_1}\cdot\boldsymbol{S}_i^{p-}/\tau)}$$



2.2. Group-level Discrimination

- **Final Loss:** $\mathcal{L} = \mathcal{L}_{\text{W-InfoNCE}} + \lambda \mathcal{L}_{\text{G-InfoNCE}}$
  where λ is a hyperparameter that controls the contribution of the two loss functions.



3. WSI retrieval method

- Due to the unique WSI characteristics, WSI-CBIR is usually implemented in two stages:
  1. Offline WSI **feature extraction**
  2. Similar WSI **searching**

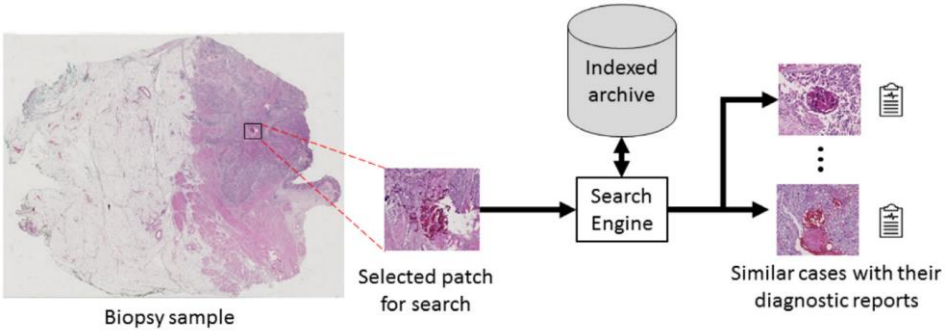


Fig. 1. General workflow of CBIR systems for digital pathology.

3. WSI retrieval method

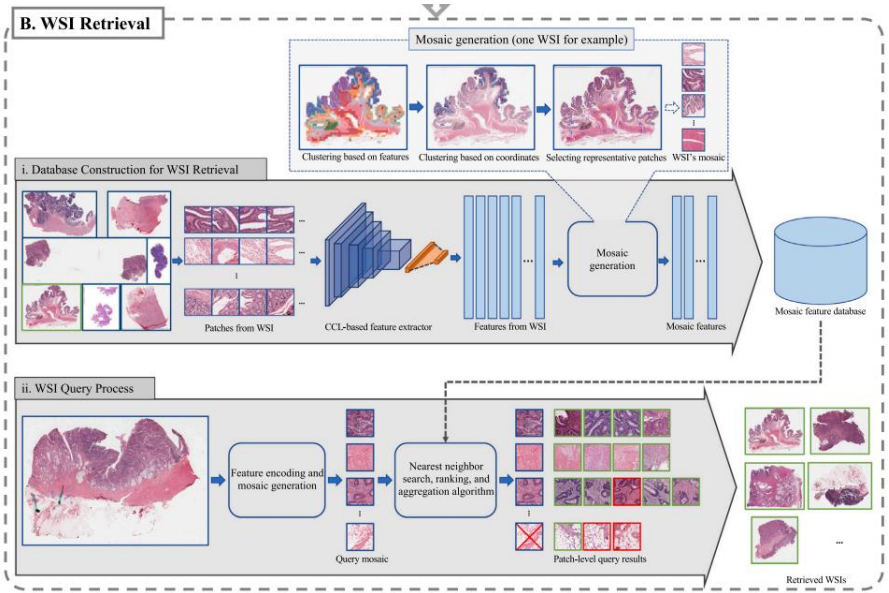
Step / Method		Yottixel [1]	FISH [2]	RetCCL
Feature Extraction	Backbone	DenseNet (ImageNet) + VQ-VAE (TCGA)	DenseNet (ImageNet)	ResNet50 (TCGA)
	Feature Compress	Binary codes	Binary codes	X **1. more accurate**
	Database Construction	Dual clustering method	Dual clustering method	Dual clustering method
WSI Searching		‘‘median-of-min’’ approach for Hamming distance nearest neighbor searching	VEB tree with an uncertainty-based ranking algorithm	Cosine-similarity-based nearest neighbor searching

Simple, but not accurate **careful parameter setting** **2. Easy to use, more accurate**

[1] Shivam Kalra, et al. "Yottixel–an image search engine for large archives of histopathology whole slide images." Medical Image Analysis. 2020.
[2] Chengkuan Chen, et al. "Fast and scalable search of whole-slide images via self-supervised deep learning." Nature Biomedical Engineering. 2022.

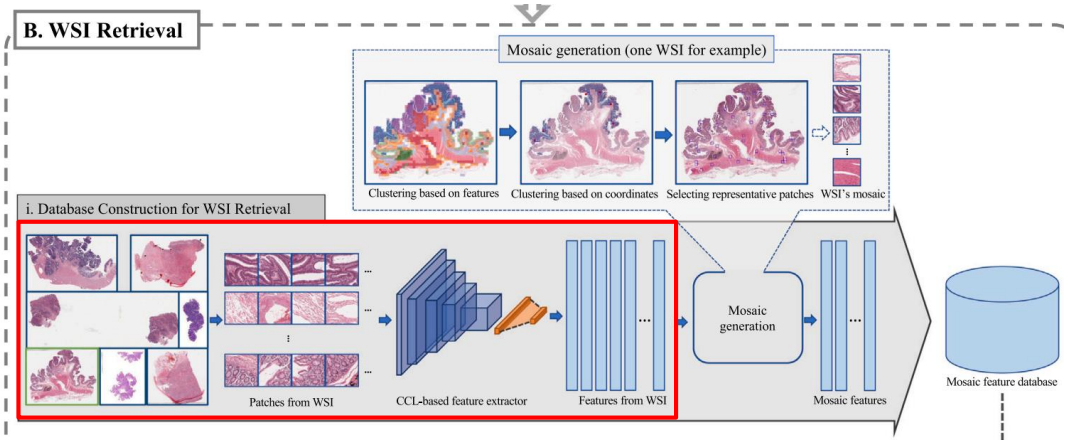
3. WSI retrieval method

- The overall procedure of our WSI retrieval contains a two-step operation:
  1. **Database Construction for WSI Retrieval**
  2. **WSI Query Process**



3.1. Database Construction for WSI Retrieval

- **Step 1.**
  - A WSI is cropped into small patches.
  - These patches are then fed into our CCL model to obtain their corresponding **feature vectors**, denoted as f_{all}.

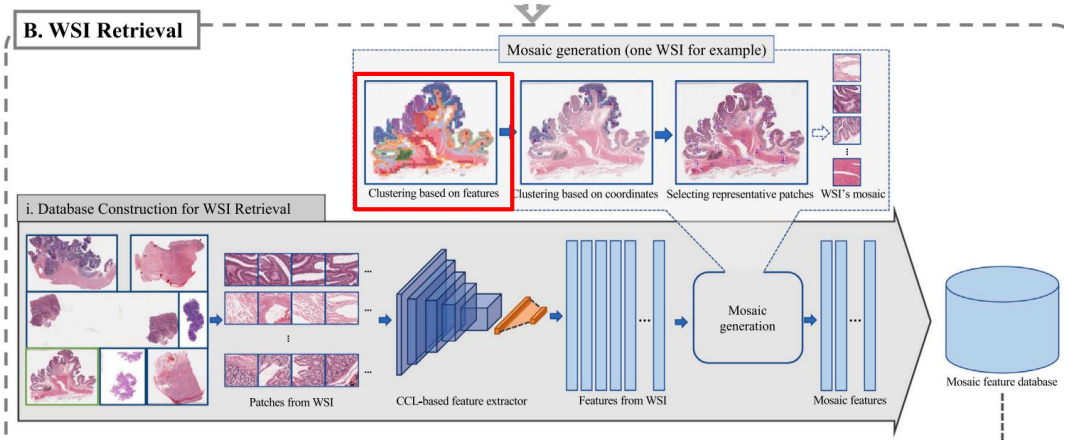


Algorithm 1 Database Construction for WSI Retrieval

$D_s \leftarrow 16$ ▷ Downsample for segmentation
$MPP \leftarrow 1.0$ ▷ Magnification for patching
$S_p \leftarrow 512$ ▷ Size of patch
$K_1 \leftarrow 9$ ▷ First clustering number
$R \leftarrow 0.2$ ▷ Second clustering ratio
$G \leftarrow \{\}$ ▷ Set of all selected features
for $I \in$ All WSIs **do**
  procedure MosaicGeneration(I, D_s, MPP, S_p, K_1, R)
    $f = \{\}$
    $S \leftarrow segment(I, D_s)$ ▷ Foreground segment for WSI
    $p \leftarrow patching(S, MPP, S_p)$ ▷ Obtain all patches
    $f_{all} \leftarrow model(p)$ ▷ Obtain all features
    $F_i \leftarrow$ FeatureKMeans(f_{all}, K_1) ▷ Feature clustering
    for $i \in K_1$ **do** $where, i = 1, 2, ..., K_1$
      $f_{rep} \leftarrow$ SpatialKMeans(F_i, R) ▷ Coordinate clustering
      $f \leftarrow f \cup f_{rep}$
    end for
    return f ▷ Return representative patch features
  end procedure
  $G \leftarrow G \cup f$
end for
return G ▷ Return mosaic database for WSI retrieval

3.1. Database Construction for WSI Retrieval

- **Step 2.**
  - Do **K-means clustering**, K_1 distinctive classes within the WSI are obtained, which are represented by $\{\boldsymbol{F}_1, \boldsymbol{F}_2, \ldots, \boldsymbol{F}_i, \ldots, \boldsymbol{F}_{K_1}\}$.

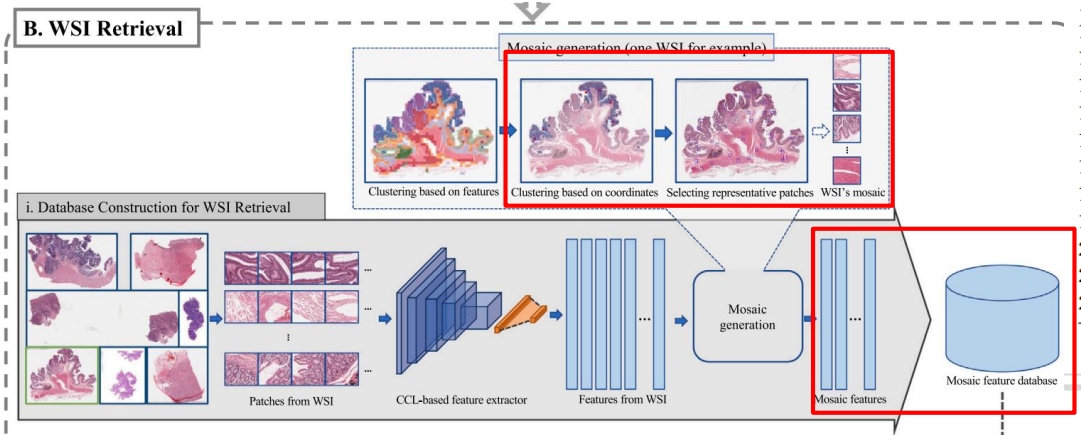


Algorithm 1 Database Construction for WSI Retrieval

$D_s \leftarrow 16$ ▷ Downsample for segmentation
$MPP \leftarrow 1.0$ ▷ Magnification for patching
$S_p \leftarrow 512$ ▷ Size of patch
$K_1 \leftarrow 9$ ▷ First clustering number
$R \leftarrow 0.2$ ▷ Second clustering ratio
$\boldsymbol{G} \leftarrow \{\}$ ▷ Set of all selected features
for $\boldsymbol{I} \in$ All WSIs **do**
  procedure MosaicGeneration($\boldsymbol{I}, D_s, MPP, S_p, K_1, R$)
    $\boldsymbol{f} = \{\}$
    $\boldsymbol{S} \leftarrow segment(\boldsymbol{I}, D_s)$ ▷ Foreground segment for WSI
    $\boldsymbol{p} \leftarrow patching(\boldsymbol{S}, MPP, S_p)$ ▷ Obtain all patches
    $\boldsymbol{f}_{all} \leftarrow model(\boldsymbol{p})$ ▷ Obtain all features
    $\boldsymbol{F}_i \leftarrow$ FeatureKMeans($\boldsymbol{f}_{all}, K_1$) ▷ Feature clustering
    for $i \in K_1$ **do** $where, i = 1, 2, ..., K_1$
      $\boldsymbol{f}_{rep} \leftarrow$ SpatialKMeans($\boldsymbol{F}_i, R$) ▷ Coordinate clustering
      $\boldsymbol{f} \leftarrow \boldsymbol{f} \cup \boldsymbol{f}_{rep}$
    end for
    return $\boldsymbol{f}$ ▷ Return representative patch features
  end procedure
  $\boldsymbol{G} \leftarrow \boldsymbol{G} \cup \boldsymbol{f}$
end for
return $\boldsymbol{G}$ ▷ Return mosaic database for WSI retrieval

3.1. Database Construction for WSI Retrieval

- **Step 3.**
  - Each cluster is further **re-clustered** into $K2$ sub-classes using their spatial coordinates as features, where $K2 = round(R \cdot n)$.
  - R : a ratio parameter and is set as 20%
  - n : the number of patches within each cluster Fi
  - Last, the patches in these final clustering **centroids** are adopted as the representation of the WSI.

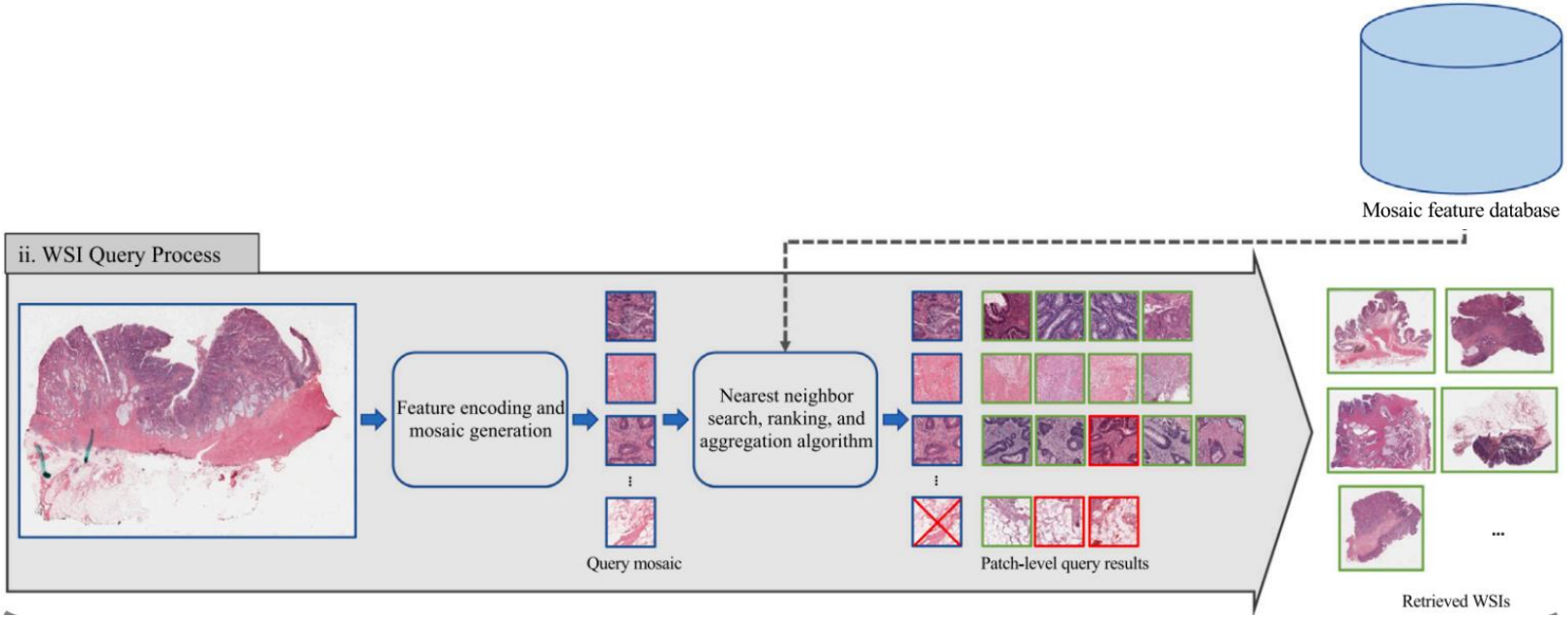


Algorithm 1 Database Construction for WSI Retrieval

$D_s \leftarrow 16$ ▷ Downsample for segmentation
$MPP \leftarrow 1.0$ ▷ Magnification for patching
$S_p \leftarrow 512$ ▷ Size of patch
$K_1 \leftarrow 9$ ▷ First clustering number
$R \leftarrow 0.2$ ▷ Second clustering ratio
$G \leftarrow \{\}$ ▷ Set of all selected features
for $I \in$ All WSIs **do**
  procedure MosaicGeneration(I, D_s, MPP, S_p, K_1, R)
    $f = \{\}$
    $S \leftarrow segment(I, D_s)$ ▷ Foreground segment for WSI
    $p \leftarrow patching(S, MPP, S_p)$ ▷ Obtain all patches
    $f_{all} \leftarrow model(p)$ ▷ Obtain all features
    $F_i \leftarrow$ FeatureKMeans(f_{all}, K_1) ▷ Feature clustering
    for $i \in K_1$ **do** $where, i = 1, 2, ..., K_1$
      $f_{rep} \leftarrow$ SpatialKMeans(F_i, R) ▷ Coordinate clustering
      $f \leftarrow f \cup f_{rep}$
    end for
    return f ▷ Return representative patch features
  end procedure
  $G \leftarrow G \cup f$
end for
return G ▷ Return mosaic database for WSI retrieval

3.2. WSI Query Process

- After building the WSI database, the subsequent WSI retrieval can be regarded as a patch-level **nearest neighbor matching**, **ranking**, and **aggregation** process.



3.2. WSI Query Process

- **Step 1.**
  - The query WSI can be represented as a mosaic with k patches, such as $WSI \quad \{P_1, P_2, \ldots, P_i \ldots, P_k\}$
  - P_i : feature vector of the ith patch
  - k : the total number of patches within the WSI.

Algorithm 2 WSI Query Process

$\boldsymbol{w}_v \leftarrow w_{v_1}, \ldots, w_{v_i}, \ldots, w_{v_n}$ ▷ Weight of each diagnosis in the database
$WSI = \{P_1, P_2, \ldots, P_i \ldots, P_k\}$ ▷ Given a query WSI with k patches
$Bag = \{\mathbb{B}_1, \mathbb{B}_2, \ldots, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ A bag contains a query patch and its retrieved patches
$\mathbb{B}_i = \{b_i^1, b_i^2, b_i^j \ldots, b_i^t\}$ ▷ Each query patch retrieves t patches
procedure CALCULATE ENTROPY FOR EACH BAG
  for $\mathbb{B}_i \in Bag$ **do** ▷ A bag $\mathbb{B}_i$ has u_i associated WSI diagnosis
    $\mathbb{D} = CosineSimilarity(P_i, \mathbb{B}_i)$ ▷ Calculate cosine similarity, where $\mathbb{D} = \{d^1, d^2, \ldots, d^j \ldots, d^t\}$
    $p_m = Probability(\boldsymbol{w}_y, \mathbb{D}, \mathbb{B}_i)$ ▷ Probability calculated for the m^{th} diagnosis occurrence within a bag
    $Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log(p_m)$ ▷ Entropy within a bag
  end for
  $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ Reorder bag by entropy
end procedure
procedure REMOVE BAGS WITH LOW QUALITY
  $\eta = \frac{1}{k} \sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$ ▷ Means of cosine similarity scores in top-5
  $Bag'' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_{k'}\}$ ▷ Remove bags with small η
end procedure
for $\mathbb{B}_i \in Bag$ **do** ▷ Vote for each diagnosis within a bag
  $\mathbb{B}_i = \{b_i^1, b_i^2, b_i^j \ldots, b_i^5\}$ ▷ Obtain the top-5 samples in each bag
  $\boldsymbol{W}_i \leftarrow \mathbb{B}_i$ ▷ Majority vote to obtain associated WSI for each bag
end for
WSIRet $= \{\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_i \ldots, \boldsymbol{W}_{k''}\}$ ▷ Find similar WSIs
return WSIRet$[1 : k]$ ▷ Return top-k similar WSIs

nearest neighbor matching

aggregation

ranking

3.2. WSI Query Process

- **Step 2.**
  - Each patch will be adopted as a query image to generate the corresponding retrieval results that are stored in k bags $\boldsymbol{Bags} = \{\mathbb{B}_1, \mathbb{B}_2, \ldots, \mathbb{B}_i \ldots, \mathbb{B}_k\}$,
  - $\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j \ldots, \boldsymbol{b}_i^t\}$ contains $\boldsymbol{t}$ retrieved patches along with their cosine similarity scores calculated with the query patch $\boldsymbol{P}_i$
  - Note that $\boldsymbol{t}$ varies for different bags.

Algorithm 2 WSI Query Process

$\boldsymbol{w}_y \leftarrow w_{y_1}, \ldots, w_{y_m}, \ldots, w_{y_U}$ ▷ Weight of each diagnosis in the database
$WSI = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_i \ldots, \boldsymbol{P}_k\}$ ▷ Given a query WSI with k patches
$Bag = \{\mathbb{B}_1, \mathbb{B}_2, \ldots, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ A bag contains a query patch and its retrieved patches
$\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j \ldots, \boldsymbol{b}_i^t\}$ ▷ Each query patch retrieves t patches
procedure CALCULATE ENTROPY FOR EACH BAG
  for $\mathbb{B}_i \in Bag$ **do** ▷ A bag $\mathbb{B}_i$ has u_i associated WSI diagnosis
    $\mathbb{D} = CosineSimilarity(\boldsymbol{P}_i, \mathbb{B}_i)$ ▷ Calculate cosine similarity, where $\mathbb{D} = \{d^1, d^2, \ldots, d^j \ldots, d^t\}$
    $p_m = Probability(\boldsymbol{w}_y, \mathbb{D}, \mathbb{B}_i)$ ▷ Probability calculated for the m^{th} diagnosis occurrence within a bag
    $Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log(p_m)$ ▷ Entropy within a bag
  end for
  $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ Reorder bag by entropy
end procedure
procedure REMOVE BAGS WITH LOW QUALITY
  $\eta = \frac{1}{k}\sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$ ▷ Means of cosine similarity scores in top-5
  $Bag'' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_{k'}\}$ ▷ Remove bags with small η
end procedure
for $\mathbb{B}_i \in Bag$ **do** ▷ Vote for each diagnosis within a bag
  $\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j \ldots, \boldsymbol{b}_i^5\}$ ▷ Obtain the top-5 samples in each bag
  $\boldsymbol{W}_i \leftarrow \mathbb{B}_i$ ▷ Majority vote to obtain associated WSI for each bag
end for
WSIRet $= \{\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_i \ldots, \boldsymbol{W}_{k''}\}$ ▷ Find similar WSIs
return WSIRet[1 : k] ▷ Return top-k similar WSIs

nearest neighbor matching

aggregation

ranking

3.2. WSI Query Process

- **Step 3.**
  - **Entropy** with the ability of uncertainty measure is used to calculate the uncertainty of each bag.

$$Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log\left(p_m\right)$$

  - u_i: the total number of diagnosis types within the bag $\mathbb{B}_i$
  - p_m: the probability of the mth diagnosis type occurring in a bag

$$p_m = \frac{\sum_{j=1}^{t} \delta(y_j, m) \cdot w_{y_j} \cdot \left(d^j + 1\right)/2}{\sum_{j=1}^{t} w_{y_j} \cdot \left(d^j + 1\right)/2}$$

where $y_j \in \{1, \ldots, m, \ldots, u_i\}$

  - $\delta(y_j, m)$: judges whether the current sample belongs to the mth diagnosis type

Algorithm 2 WSI Query Process

$\boldsymbol{w}_y \leftarrow w_{y_1}, \ldots, w_{y_m}, \ldots, w_{y_U}$ ▷ Weight of each diagnosis in the database
$WSI = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_i \ldots, \boldsymbol{P}_k\}$ ▷ Given a query WSI with k patches
$Bag = \{\mathbb{B}_1, \mathbb{B}_2, \ldots, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ A bag contains a query patch and its retrieved patches
$\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j \ldots, \boldsymbol{b}_i^t\}$ ▷ Each query patch retrieves t patches
procedure CALCULATE ENTROPY FOR EACH BAG
  for $\mathbb{B}_i \in Bag$ **do** ▷ A bag $\mathbb{B}_i$ has u_i associated WSI diagnosis
    $\mathbb{D} = CosineSimilarity(\boldsymbol{P}_i, \mathbb{B}_i)$ ▷ Calculate cosine similarity, where $\mathbb{D} = \{d^1, d^2, \ldots, d^j \ldots, d^t\}$
    $p_m = Probability(\boldsymbol{w}_y, \mathbb{D}, \mathbb{B}_i)$ ▷ Probability calculated for the m^{th} diagnosis occurrence within a bag
    $Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log\left(p_m\right)$ ▷ Entropy within a bag
  end for
  $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_k\}$ ▷ Reorder bag by entropy
end procedure
procedure REMOVE BAGS WITH LOW QUALITY
  $\eta = \frac{1}{k}\sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$ ▷ Means of cosine similarity scores in top-5
  $Bag'' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i \ldots, \mathbb{B}_{k'}\}$ ▷ Remove bags with small η
end procedure
for $\mathbb{B}_i \in Bag$ **do** ▷ Vote for each diagnosis within a bag
  $\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j \ldots, \boldsymbol{b}_i^5\}$ ▷ Obtain the top-5 samples in each bag
  $\boldsymbol{W}_i \leftarrow \mathbb{B}_i$ ▷ Majority vote to obtain associated WSI for each bag
end for
WSIRet $= \{\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_i \ldots, \boldsymbol{W}_{k''}\}$ ▷ Find similar WSIs
return WSIRet$[1 : k]$ ▷ Return top-k similar WSIs

nearest neighbor matching

aggregation

ranking

3.2. WSI Query Process

- **Step 4.**
  - **Reorder** these bags in descending order based on their entropy to obtain new bags $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i ..., \mathbb{B}_k\}$
- **Step 5.**
  - **Remove** bags whose average cosine similarity scores are smaller than the η.

    $$\eta = \frac{1}{k} \sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$$

  - ***AveTop***: the average of the top-5 cosine similarity scores within the bag.

Algorithm 2 WSI Query Process

$\boldsymbol{w}_y \leftarrow w_{y_1}, ..., w_{y_m}, ..., w_{y_U}$ ▷ Weight of each diagnosis in the database
$WSI = \{\boldsymbol{P}_1, \boldsymbol{P}_2, ..., \boldsymbol{P}_i ..., \boldsymbol{P}_k\}$ ▷ Given a query WSI with k patches
$Bag = \{\mathbb{B}_1, \mathbb{B}_2, ..., \mathbb{B}_i ..., \mathbb{B}_k\}$ ▷ A bag contains a query patch and its retrieved patches
$\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j ..., \boldsymbol{b}_i^t\}$ ▷ Each query patch retrieves t patches
procedure CALCULATE ENTROPY FOR EACH BAG
  for $\mathbb{B}_i \in Bag$ **do** ▷ A bag $\mathbb{B}_i$ has u_i associated WSI diagnosis
    $\mathbb{D} = CosineSimilarity(\boldsymbol{P}_i, \mathbb{B}_i)$ ▷ Calculate cosine similarity, where $\mathbb{D} = \{d^1, d^2, ..., d^j ..., d^t\}$
    $p_m = Probability(\boldsymbol{w}_y, \mathbb{D}, \mathbb{B}_i)$ ▷ Probability calculated for the m^{th} diagnosis occurrence within a bag
    $Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log(p_m)$ ▷ Entropy within a bag
  end for
  $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i ..., \mathbb{B}_k\}$ ▷ Reorder bag by entropy
end procedure
procedure REMOVE BAGS WITH LOW QUALITY
  $\eta = \frac{1}{k} \sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$ ▷ Means of cosine similarity scores in top-5
  $Bag'' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i ..., \mathbb{B}_{k'}\}$ ▷ Remove bags with small η
end procedure
for $\mathbb{B}_i \in Bag$ **do** ▷ Vote for each diagnosis within a bag
  $\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j ..., \boldsymbol{b}_i^5\}$ ▷ Obtain the top-5 samples in each bag
  $\boldsymbol{W}_i \leftarrow \mathbb{B}_i$ ▷ Majority vote to obtain associated WSI for each bag
end for
WSIRet $= \{\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_i ..., \boldsymbol{W}_{k''}\}$ ▷ Find similar WSIs
return WSIRet[1 : k] ▷ Return top-k similar WSIs

nearest neighbor matching

aggregation

ranking

3.2. WSI Query Process

- **Step 6.**
  - **Majority vote** to obtain associated WSI for each bag according to **top-5** samples in each bag.
- **Step 7.**
  - Return top-k similar WSIs

Algorithm 2 WSI Query Process

$\boldsymbol{w}_y \leftarrow w_{y_1}, ..., w_{y_m}, ..., w_{y_U}$ ▷ Weight of each diagnosis in the database
$WSI = \{\boldsymbol{P}_1, \boldsymbol{P}_2, ..., \boldsymbol{P}_i ..., \boldsymbol{P}_k\}$ ▷ Given a query WSI with k patches
$Bag = \{\mathbb{B}_1, \mathbb{B}_2, ..., \mathbb{B}_i ..., \mathbb{B}_k\}$ ▷ A bag contains a query patch and its retrieved patches
$\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j ..., \boldsymbol{b}_i^t\}$ ▷ Each query patch retrieves t patches
procedure CALCULATE ENTROPY FOR EACH BAG
  for $\mathbb{B}_i \in Bag$ **do** ▷ A bag $\mathbb{B}_i$ has u_i associated WSI diagnosis
    $\mathbb{D} = CosineSimilarity(\boldsymbol{P}_i, \mathbb{B}_i)$ ▷ Calculate cosine similarity, where $\mathbb{D} = \{d^1, d^2, ..., d^j ..., d^t\}$
    $p_m = Probability(\boldsymbol{w}_y, \mathbb{D}, \mathbb{B}_i)$ ▷ Probability calculated for the m^{th} diagnosis occurrence within a bag
    $Ent_i = -\sum_{m=1}^{u_i} p_m \cdot \log(p_m)$ ▷ Entropy within a bag
  end for
  $Bag' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i ..., \mathbb{B}_k\}$ ▷ Reorder bag by entropy
end procedure
procedure REMOVE BAGS WITH LOW QUALITY
  $\eta = \frac{1}{k}\sum_{i=1}^{k} AveTop\{\mathbb{B}_i\}$ ▷ Means of cosine similarity scores in top-5
  $Bag'' = \{\mathbb{B}_1, \mathbb{B}_2, \mathbb{B}_i ..., \mathbb{B}_{k'}\}$ ▷ Remove bags with small η
end procedure
for $\mathbb{B}_i \in Bag$ **do** ▷ Vote for each diagnosis within a bag
  $\mathbb{B}_i = \{\boldsymbol{b}_i^1, \boldsymbol{b}_i^2, \boldsymbol{b}_i^j ..., \boldsymbol{b}_i^5\}$ ▷ Obtain the top-5 samples in each bag
  $\boldsymbol{W}_i \leftarrow \mathbb{B}_i$ ▷ Majority vote to obtain associated WSI for each bag
end for
WSIRet $= \{\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{W}_i ..., \boldsymbol{W}_{k''}\}$ ▷ Find similar WSIs
return WSIRet$[1 : k]$ ▷ Return top-k similar WSIs

nearest neighbor matching

aggregation

ranking

4.1. Datasets

Dataset	# of WSIs	# of types	magnification	patchsize	# of patches
TCGA	29,763	32	20×	1024 × 1024	14,325,848
PAIP	2,457	6	20×	1024 × 1024	1,254,414
UniToPatho	292	4	20×	1812 × 1812	8,699
TissueNet	1,016	4	-	1200 × 1200	5,926
DiagSet-A.2.	430	4	5x	224 × 224	68,562

4.2. Evaluation metrics

- For image retrieval
  - ***ACC@k:*** *ACC@k* = 1 if any one of the top-k returns has the same label as the query image
  - ***mMV@k:*** *mMV@k* = 1 only if the majority of these retrieved images have the same label as the query image
- For the downstream classification task
  - Accuracy (ACC)
  - F1 score
- All the image retrieval validation experiments are conducted using the leave-one-patient-out strategy to avoid information leakage due to the occasional existence of multiple WSIs from the same patient.

4.3. Results of ablation experiments

- Effect of network components

Table 1
Ablation results on TissueNet and UniToPatho datasets.

	TissueNet				UniToPatho			
	Acc@1	*Acc*@3	*Acc*@5	*mMV*@5	*Acc*@1	*Acc*@3	*Acc*@5	*mMV*@5
ImageNet	50.35	77.62	87.68	46.15	58.17	82.89	89.45	59.01
MoCo v2	64.74	86.27	92.77	65.57	63.36	83.38	89.57	64.86
MoCo v2+Gro.	66.20	87.07	93.10	67.56	65.49	83.95	90.04	66.63
MoCo v2+Mem.	66.64	87.21	93.12	68.78	65.87	84.10	90.08	67.19
MoCo v2+Gro.+Mem. (Ours)	**67.09**	**87.81**	**93.40**	**70.01**	**66.55**	**84.32**	**90.31**	**68.35**

4.3. Results of ablation experiments

- Effect of different number of clustering centers (Q and S)

Table 2
Effect of different number of Q values on retrieval accuracy using the TissueNet and UniToPatho datasets.

Q	TissueNet		UniToPatho	
	$Acc@1$	$mMV@5$	$Acc@1$	$mMV@5$
15	66.37	68.78	66.07	67.60
20	66.86	69.25	66.18	67.83
25	**67.09**	**70.01**	**66.55**	**68.35**
30	66.99	69.63	66.28	67.69
35	66.82	69.02	65.98	67.49

Table 3
Effect of different number of S values on retrieval accuracy using the TissueNet and UniToPatho datasets.

S	TissueNet		UniToPatho	
	$Acc@1$	$mMV@5$	$Acc@1$	$mMV@5$
20	66.33	68.29	65.56	67.10
25	66.76	69.12	65.91	67.33
30	**67.09**	**70.01**	**66.55**	**68.35**
40	66.96	69.39	66.01	67.61

4.3. Results of ablation experiments

- Effect of different number of MLP heads

Table 4
Effect of different number of MLP heads on retrieval accuracy using the TissueNet and UniToPatho datasets.

	TissueNet		UniToPatho	
	Acc@1	*mMV*@5	*Acc*@1	*mMV*@5
One head	65.25	66.94	64.09	65.69
Two heads	**67.09**	**70.01**	**66.55**	**68.35**

4.4. Comparison between our CCL and other SSL-based feature extractors

- Compare with SimCLR v1, SwAV, and MoCo v2

Table 6
Patch-level retrieval results by comparing our CCL with other SSL-based feature extractors.

	TissueNet				UniToPatho			
	Acc@1	*Acc*@3	*Acc*@5	*mMV*@5	*Acc*@1	*Acc*@3	*Acc*@5	*mMV*@5
SimCLR v1 (Chen et al., 2020b)	62.60	85.53	92.85	65.04	61.12	83.25	89.50	62.08
MoCo v2 (Chen et al., 2020a)	64.74	86.27	92.77	65.57	63.36	83.38	89.54	64.86
SwAV (Caron et al., 2020)	65.39	86.06	92.54	66.67	64.18	83.45	89.78	64.98
Ours	**67.09**	**87.81**	**93.40**	**70.01**	**66.55**	**84.32**	**90.31**	**68.35**

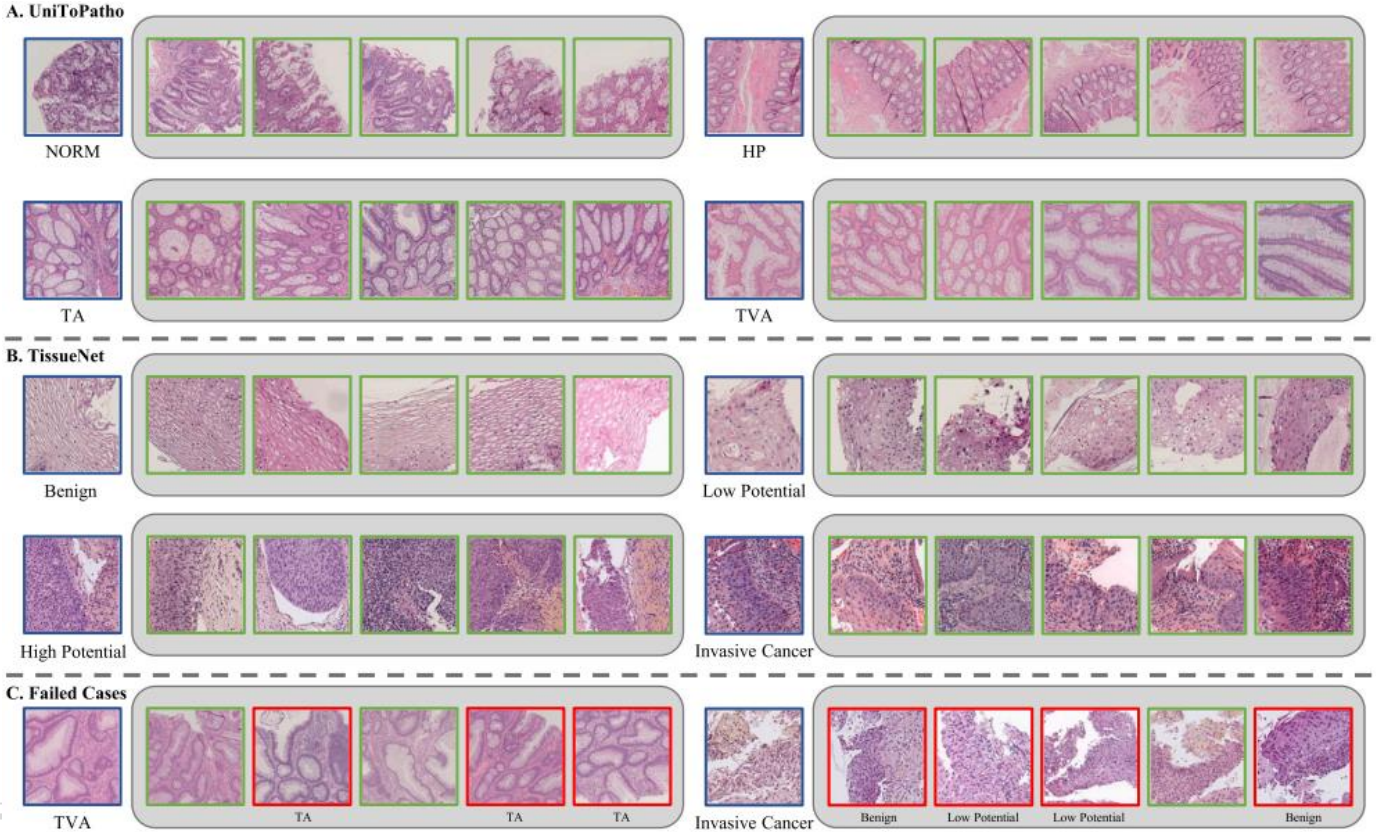
4.4. Results of ablation experiments

- Effect of different settings of w
  - This may be due to the fact that the instance discrimination task in contrastive learning needs more explicit pseudo-labels to supervise network training.

Table 5
Effect of different settings of w on retrieval accuracy using the TissueNet and UniToPatho datasets.

w	TissueNet		UniToPatho	
	$Acc@1$	$mMV@5$	$Acc@1$	$mMV@5$
0.1	66.71	69.61	66.13	67.69
0.2	**67.09**	**70.01**	**66.55**	**68.35**
0.5	66.52	68.97	65.79	67.15
1	66.20	67.56	65.49	66.63
Soft	66.23	67.85	65.44	66.81
Hard (w=0.2)	**67.09**	**70.01**	**66.55**	**68.35**

4.5. Interpretability analysis for patch-level retrieval



4.6. Results of WSI anatomic site retrieval

Table 7
Results of anatomical site retrieval experiment on frozen WSIs in terms of $mMV@10$.

Anatomic Sites	#WSI	#Patient	$mMV@10$	
			Yottixel	Ours
Brain	1818	1091	83.86	**90.21**
Endocrine	796	769	35.37	**80.44**
Gastrointestinal	1984	1234	62.86	**81.77**
Gynecologic	2284	1502	**68.86**	50.74
Hematopoiesis	182	170	45.85	**66.49**
Melanocytic	542	536	37.20	**47.42**
Liver/PB	669	610	35.35	**76.23**
Pulmonary	1658	1093	59.30	**79.99**
Urinary	2035	1323	64.59	**79.35**
Prostate/Testis	759	639	68.07	**84.28**
Breast	1520	1080	66.35	**91.35**
Mesenchymal	263	260	11.19	**74.13**
Head and Neck	727	471	26.24	**79.64**
Macro-average	–	–	51.16	**75.54**
Weighted-average	–	–	60.45	**75.50**

4.6. Results of WSI anatomic site retrieval

Table 8
Results of anatomical site retrieval experiment on FFPE WSIs in terms of $mMV@10$.

Anatomic Sites	#WSI	#Patient	$mMV@10$		
			Yottixel	FISH	Ours
Brain	1699	878	91.37	**95.80**	93.41
Endocrine	942	737	**73.93**	70.00	69.64
Gastrointestinal	1205	1148	65.12	56.10	**83.80**
Gynecologic	1074	933	63.71	69.40	**76.82**
Hematopoiesis	224	165	52.03	79.40	**80.36**
Melanocytic	554	512	37.20	48.60	**53.97**
Liver/PB	628	586	63.75	72.50	**89.97**
Pulmonary	1137	1028	75.83	71.60	**81.60**
Urinary	1394	1280	66.01	54.20	**69.80**
Prostate/Testis	703	552	80.31	84.40	**86.49**
Breast	1160	1045	70.87	75.80	**93.71**
Mesenchymal	599	254	50.70	61.70	**91.65**
Head and Neck	472	450	49.14	51.40	**77.97**
Macro-average	–	–	64.61	68.53	**80.71**
Weighted-average	–	–	69.05	70.17	**81.62**

4.7. Results of WSI cancer subtype retrieval

Table 9
Results of cancer subtype retrieval experiment on frozen WSIs in terms of $mMV@5$.

WSI Type	#WSI	#Patient	$mMV@5$		WSI Type	#WSI	#Patient	$mMV@5$	
			Yottixel	Ours				Yottixel	Ours
Pulmonary					**Liver/PB**				
LUAD	822	505	68.23	**78.10**	*CHOL*	51	51	35.29	**39.22**
LUSC	751	486	78.25	**90.28**	*LIHC*	398	375	94.36	**94.97**
MESO	87	87	27.71	**83.91**	*PAAD*	218	184	**91.66**	90.83
Urinary					**Gynecologic**				
BLCA	429	410	92.85	**98.37**	*UCEC*	711	542	**90.07**	81.86
KIRC	1088	536	**97.81**	93.75	*CESC*	309	302	64.42	**78.32**
KICH	146	90	78.26	**91.78**	*UCS*	57	57	10.20	**68.42**
KIRP	375	281	62.12	**88.80**	*OV*	1203	589	**99.07**	93.43
Gastrointestinal					**Endocrine**				
COAD	855	459	**63.73**	55.55	*ACC*	91	91	45.67	**81.32**
ESCA	173	172	25.90	**67.05**	*PCPG*	180	175	85.63	**88.89**
READ	330	171	14.32	**37.27**	*THCA*	538	502	97.08	**98.33**
STAD	632	432	71.10	**73.42**					
Melanocytic					**Prostate/Testis**				
UVM	69	69	46.37	**88.41**	*TGCT*	155	149	86.45	**98.06**
SKCM	470	467	**98.70**	94.68	*PRAD*	605	489	**98.33**	98.18
Brain					**Hematopoiesis**				
GBM	1097	577	**94.19**	85.78	*DLBC*	59	46	**91.22**	77.97
LGG	715	509	82.58	**86.30**	*THYM*	124	124	**97.58**	96.77

The macro-average $mMV@5$ of Yottixel and our method are 72.04 and 82.76, respectively.

4.7. Results of WSI cancer subtype retrieval

Table 10
Results of cancer subtype retrieval experiment on FFPE WSIs in terms of $mMV@5$.

WSI Type	#WSI	#Patient	$mMV@5$			WSI Type	#WSI	#Patient	$mMV@5$		
			Yottixel	FISH	Ours				Yottixel	FISH	Ours
Pulmonary						**Liver/PB**					
LUAD	538	475	70.96	79.81	**84.01**	*CHOL*	38	38	43.58	46.15	**55.26**
LUSC	512	478	81.70	71.68	**84.18**	*LIHC*	381	365	93.65	90.30	**96.06**
MESO	87	75	8.13	55.81	**72.41**	*PAAD*	203	183	91.04	89.47	**96.55**
Urinary						**Gynecologic**					
BLCA	457	385	95.81	93.22	**98.03**	*UCEC*	595	506	**92.22**	84.28	84.87
KIRC	519	513	91.66	92.29	**93.06**	*CESC*	285	268	62.45	78.78	**86.67**
KICH	109	109	75.92	90.10	**95.41**	*UCS*	87	53	42.22	71.26	**72.41**
KIRP	297	273	67.22	66.33	**90.91**	*OV*	107	106	66.98	**83.18**	70.09
Gastrointestinal						**Endocrine**					
COAD	469	447	**76.14**	48.30	69.72	*ACC*	226	56	93.83	**96.04**	94.69
ESCA	158	156	59.87	79.75	**82.28**	*PCPG*	195	175	88.77	**91.84**	82.99
READ	169	161	10.19	**44.94**	25.44	*THCA*	521	506	97.66	98.07	**99.04**
STAD	409	384	74.23	74.23	**76.53**						
Melanocytic						**Prostate/Testis**					
UVM	80	80	83.75	70.00	**97.50**	*TGCT*	254	149	**99.21**	97.64	97.64
SKCM	474	432	99.57	**99.58**	97.89	*PRAD*	449	403	98.43	98.44	**98.66**
Brain						**Hematopoiesis**					
GBM	857	387	**91.88**	87.75	81.43	*DLBC*	44	44	58.13	**88.37**	72.73
LGG	842	491	89.77	**97.02**	83.73	*THYM*	180	121	98.87	93.89	**98.89**

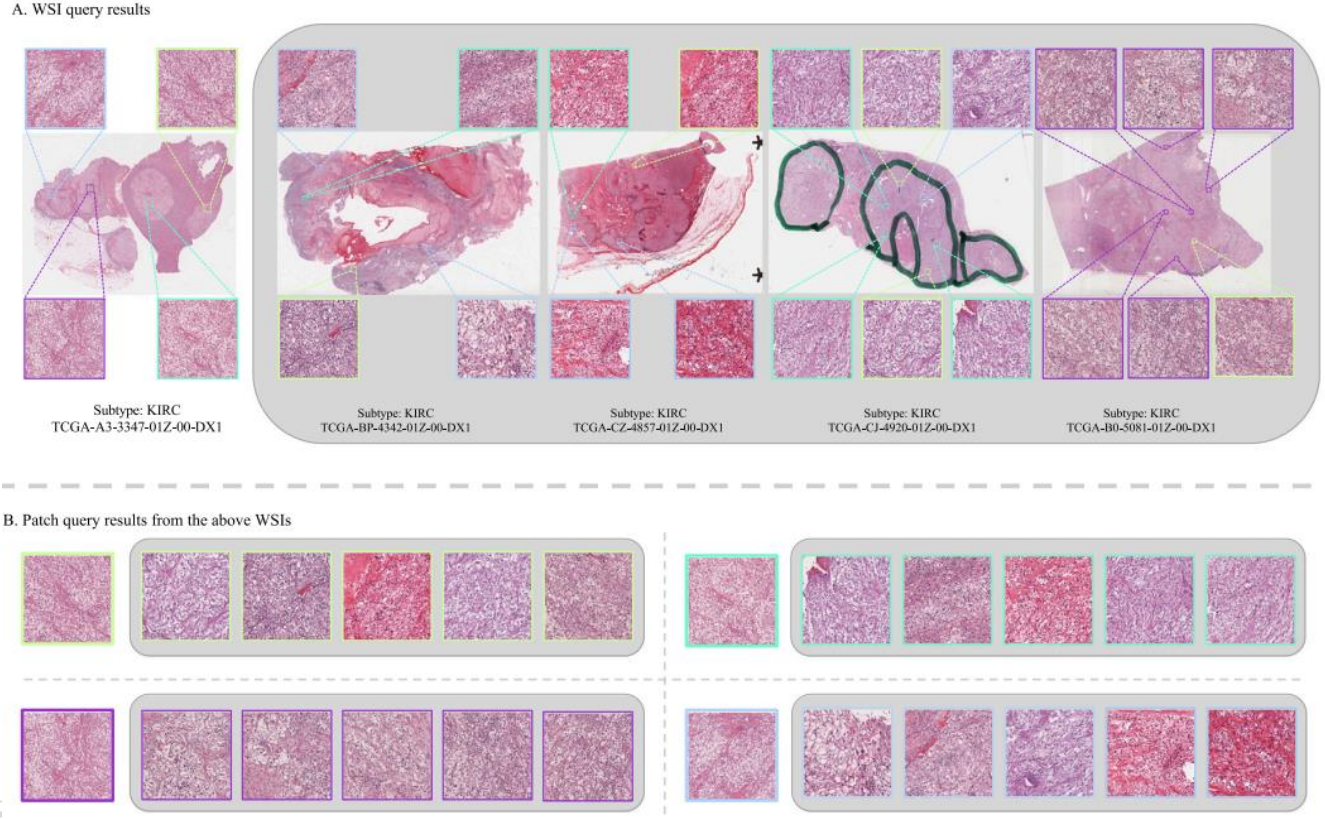
The macro-average $mMV@5$ of Yottixel, FISH, and our method are 75.99, 81.33, and 84.11, respectively.

4.8. Results of different feature extraction methods

Table 1 | Results of anatomical site retrieval accuracy on FFPE WSIs ($mMV@10$) using different feature extraction methods. The columns of Yottixel and FISH are copied directly from their publications. CCL+Yottixel and CCL+FISH are obtained by replacing the two features (ImageNet features and the color histogram features) used in both the Yottixel and FISH methods with our CCL-based features.

Anatomic Sites	#WSI	#Patient	$mMV@10$				
			Yottixel	CCL+Yottixel	FISH	CCL+FISH	RetCCL (ours)
Brain	1699	878	91.37	93.22	95.80	**96.05**	93.41
Endocrine	942	737	**73.93**	77.37	70.00	73.01	69.64
Gastrointestinal	1205	1148	65.12	75.20	56.10	77.18	**83.80**
Gynecologic	1074	933	63.71	66.81	69.40	71.59	**76.82**
Hematopoiesis	224	165	52.03	63.39	79.40	73.66	**80.36**
Melanocytic	554	512	37.20	43.18	48.60	52.09	**53.97**
Liver/PB	628	586	63.75	71.47	72.50	83.12	**89.97**
Pulmonary	1137	1028	75.83	73.47	71.60	77.92	**81.60**
Urinary	1394	1280	66.01	67.51	54.20	67.15	**69.80**
Prostate/Testis	703	552	80.31	83.25	84.40	**87.86**	86.49
Breast	1160	1045	70.87	78.20	75.80	88.75	**93.71**
Mesenchymal	599	254	50.70	64.27	61.70	78.53	**91.65**
Head and Neck	472	450	49.14	61.23	51.40	74.23	**77.97**
Macro-average	-	-	64.61	70.66	68.53	77.01	**80.71**
Weighted-average	-	-	69.05	73.79	70.17	78.68	**81.62**

4.9. Interpretability analysis for WSI-level retrieval



4.10. Results of downstream classification task

- Backbone model: **ResNet50**
- ImageNet: ImageNet pretrained features in a supervised manner
- A supervised baseline using 100% of the training data is also implemented, which produces an ACC of 0.8482 and a F1 score of 0.8462

Table 11
Linear evaluation results on DiagSet-A.2 dataset with different sizes of training data. All these methods adopt ResNet50 as the backbone model. The ImageNet means ImageNet pretrained features in a supervised manner. Note that a supervised baseline using 100% of the training data is also implemented, which produces an ACC of 0.8482 and a F1 score of 0.8462.

Methods	Percentage of training data									
	2%		5%		10%		20%		50%	
	ACC	F1	ACC	F1	ACC	F1	ACC	F1	ACC	F1
ImageNet	0.7686	0.7057	0.7764	0.7512	0.7842	0.7600	0.7885	0.7655	0.7969	0.7749
SimCLR v1	0.7862	0.7370	0.7962	0.7725	0.8176	0.7903	0.8233	0.8041	0.8266	0.8129
MoCo v2	0.7955	0.7419	0.8066	0.7929	0.8248	0.8119	0.8298	0.8124	0.8368	0.8198
SwAV	0.7970	0.7506	0.8211	0.8056	0.8354	0.8211	0.8408	0.8236	0.8471	0.8308
CCL (ours)	**0.8086**	**0.7617**	**0.8381**	**0.8136**	**0.8461**	**0.8301**	**0.8536**	**0.8467**	**0.8563**	**0.8469**

5. Conclusion

- **RetCCL** for both WSI-level and patch-level retrieval
  - Novel **CCL-based feature extractor**.
  - A ranking and aggregation algorithm for WSI retrieval.
- RetCCL outperforms existing WSI retrieval methods by a large margin.
- Our CCL-based feature is also superior to the ImageNet pretrained feature or other SSL-based features

Thanks For Listening !