# Manifold Learning - Based Image Retrieval

#1

## I: Feature Extraction and Similarity Computing:

1. The image content is encoded through the utilization of a feature-extraction process.

2. Let $D$ be an image descriptor defined as a tuple $(\epsilon, \delta)$ where:

   * $\boxed{\epsilon : O_i \longrightarrow \mathbb{R}^d}$ is a function which extracts a feature vector $\underline{v_i}$ from the image object $O_i$

   * $\boxed{\delta : \mathbb{R}^d \times \mathbb{R}^d \longrightarrow \mathbb{R}^+}$ is a function we computes the distance between two image objects $O_i$ and $O_j$.

   * Thus, we may write that: $\boxed{\begin{cases} \epsilon(O_i) = \underline{v_i} \in \mathbb{R}^d \\ \epsilon(O_j) = \underline{v_j} \in \mathbb{R}^d \end{cases}}$ and

$$\boxed{\delta(\epsilon(O_i), \epsilon(O_j)) = \delta(v_i, v_j) = \| \underline{v_i} - \underline{v_j} \|}$$

   * The similarity measure $\delta(\cdot,\cdot)$ can be used for ranking tasks, by forming a fundamental component of a similarity measure $P(O_i, O_j)$. $\left[ P(O_i, O_j) \sim \frac{1}{\| v_i - v_j \|} \right]$

## II: Image Retrieval and Rank Model:

1. Let $C = \{O_1, O_2, \dots, O_n\}$ be a collection of images.

2. The target of image retrieval refers to the process of determining image objects from $C$ based on their content.

3. That is, given a query object $O_q \in C$, obtain a ranked list $\boxed{\tau_q = (O_1, O_2, \dots, O_n)}$ as a response based on the similarity measure $P(\cdot,\cdot)$

4. • The ranked list $\tau_q$ is a permutation of the image set $\mathcal{C}$.

• Thus, $\tau_q$ may be defined as a bijection from the set $\mathcal{C}$ to the set $[n] = \{1, 2, \ldots, n\}$:

$$\boxed{\tau_q : \mathcal{C} \to [n]}$$

• In this setting, $\tau_q(i)$ indicates that object $o_i$ is the i-th most similar object according to the ranked list induced by object $o_q$.

• Moreover, if $\boxed{\tau_q(i) < \tau_q(j) \rightarrow \rho(o_q, o_i) \geq \rho(o_q, o_j)}$

5. $\tau_q$ can be very expensive to compute, especially when $n$ is very high. Therefore, the computed ranked lists may be constrained to consider only a subset $\mathcal{C}_L \subset \mathcal{C}$ from the complete image set. $(|\mathcal{C}_L| = L)$

In this context, $\boxed{\tau_q : \mathcal{C}_L \to [L] \text{ where } L \ll n.}$

6. Every object $o_i \in \mathcal{C}$ can be taken as a query object $o_q$ in order to obtain a set of ranked lists:

$$\boxed{\mathcal{T} = \{\tau_1, \tau_2, \ldots, \tau_n\}}$$

7. Based on the ranked list $\tau_q$ of each query object, we can define its k-nearest neighbours as:

$$\boxed{N_k(q) = \{\tau_q(1), \tau_q(2), \ldots, \tau_q(k)\}}$$

## III: Manifold Ranking:

- The main objective of your assignment is to exploit the similarity information encoded in the set of ranked list $\mathcal{T}$, capturing the manifold structure of the dataset.

- That is, obtain a more effective set of ranked lists $\mathcal{T}_r$

$$\boxed{\mathcal{T}_r = f(\mathcal{T})}$$

in an unsupervised learning setting.

## IV: Hypergraph Manifold Ranking:

A. Rank Normalization
B. Hypergraph Construction
C. Hyperedge Similarities
D. Cartesian Product of Hyperedge Elements
E. Hypergraph-Based Similarity.

## A: Rank Normalization

1 We may obtain a symmetric (normalized) similarity measure $P_n(\cdot,\cdot)$ between a pair of images $o_i$ and $o_j$, as:

$$\boxed{P_n(o_i, o_j) = 2L - (\tau_i(j) + \tau_j(i))}$$

2 $P_n(\cdot,\cdot)$ is a symmetric similarity measure since:

$$\boxed{P_n(o_i, o_j) = P_n(o_j, o_i)}$$

We could define a symmetric ranking measure of the form $\hat{\tau}_i(j) = \hat{\tau}_j(i) = \frac{\tau_i(j) + \tau_j(i)}{2}$

## B: Hypergraph Construction

1: • A hypergraph is a generalization of a traditional graph, in which the edges are non-empty subsets of the vertex set and, therefore, can connect any number of vertices.

2: • Let $V = \{v_1, v_2, \dots, v_n\}$ be a finite set of vertices and

• $E$ is a family of subsets of $V$ such that

$$\boxed{\bigcup_{e \in E} = V}$$

which denotes the set of hyperedges.

• In this context, the hypergraph is defined as the triplet:

$$\boxed{G = (V, E, w)}$$

• Each hyperedge $e_i \subset V$ is associated with a positive weight $w(e_i) \geq 0$, which denotes the confidence of the relationships established by the hyperedge $e_i$.

3: Each vertice $v_i \in V$ is associated with an object $o_i \in \mathcal{O}$.

4: A hyperedge $e_i \in E$ is said to incident with a vertex $v_j \in V$ when $v_j \in e_i$.

- In this way, a hypergraph may be represented as an $\underline{\underline{H}}_b$ $|E| \times |V|$ incident matrix as:

$$\boxed{h_b(e_i, v_j) = \begin{cases} 1, & v_j \in e_i; \\ 0, & \text{otherwise}. \end{cases}}$$

5: A hyperedge $e_i$ may be defined as a subset of vertices $e_i = \{v_1, v_2, \ldots, v_m\}$. Matrix $\underline{H}_b$ allows only for a binary assignment of a vertex to a hyperedge.

6: A probabilistic hypergraph model may be defined by considering a $|V| \times |V|$ affinity matrix $\underline{\underline{W}}$ over $V$ such that $W(i,j) \in [0, 1]$.

7: Given that $\boxed{v_i \in V \longleftrightarrow o_i \in \mathcal{O}}$, we may define each hyperedge $e_i$ as:

$$\boxed{e_i = \{v_j \in V : o_j \in N_K(o_i)\}}$$

$W(i,j) = w(e_i, v_j)$

8: Given that $\boxed{w(i,j) = 1 - \log_{x+1}[z_i(j)]}$ a non-binary version $\underline{\underline{H}}$ of the original incident matrix can be defined as:

$$\boxed{h(e_i, v_j) = \begin{cases} w(e_i, v_j), & v_j \in e_i; \\ 0, & \text{otherwise}. \end{cases}}$$

9: Each hyperedge $e_i$ is formed by the K-th nearest neighbours of $O_i$ including $O_i$. Thus, each hyperedge contains exactly K vertices.

(A) Since each $e_i$ is defined relative to each $v_i$, it is easy to deduce that $|E| = |V|$

10: The weight assigned to hyperedge $e_i$ may be computed as

$$\boxed{W(e_i) = \sum_{v_j \in e_i} W(e_i, v_j)}$$

## C: Hyperedges Similarities

1: Compute similarity between any pair of hyperedges $e_i$ and $e_j$ as:

$$\boxed{\underbrace{\underline{\underline{S_h}}}_{|E|\times|E|} = \underbrace{\underline{\underline{H}}}_{|E|\times|V|} \cdot \underbrace{\underline{\underline{H}}^T}_{(|V|\times|E|)}}$$

2: Compute similarity between any pair of vertices $v_i$ and $v_j$ as:

$$\boxed{\underbrace{\underline{\underline{S_v}}}_{|V|\times|V|} = \underbrace{\underline{\underline{H}}^T}_{|V|\times|E|} \underbrace{\underline{\underline{H}}}_{|E|\times|V|}}$$

3: Compute combined hyperedge and vertices similarity:

$$\boxed{\underbrace{\underline{\underline{S}}}_{|V|\times|V|} = \underline{\underline{S_h}} \circ \underline{\underline{S_v}}}$$

Hadamard product
↓
Element-wise Multiplication

# D: Cartesian Product of Hyperedge Elements

1: Given two hyperedges $e_q$ and $e_\ell$, the Cartesian product between them can be defined as:

$$\boxed{e_q \times e_\ell = \{ (v_x, v_y) : v_x \in e_q \wedge v_y \in e_\ell \}}$$

2: For each pair of vertices $(v_i, v_j) \in e_q^2$ a pairwise similarity relationship $p : E \times V \times V \longrightarrow \mathbb{R}^+$ as

$$\boxed{p(e_q, v_i, v_j) = w(e_q) * \overset{n}{\uparrow} w(e_q, v_i) * \overset{n}{\uparrow} w(e_q, v_j)}$$

3: A similarity measure can be defined based on the Cartesian product for any pair of vertices irrespective of the hyperedge they belong to: $\longrightarrow \underline{\underline{C}} \in M_{|V| \times |V|}$

$$\boxed{C(v_i, v_j) = \sum_{e_q \in E \wedge (v_i, v_j) \in e_q^2} p(e_q, v_i, v_j)}$$

$\underline{\underline{4}}$: Define a new overall weight matrix $\underline{\underline{\hat{W}}} \in M_{|V| \times |V|}$ for each pair of vertices, as:

$$\boxed{\underset{|V| \times |V|}{\underline{\underline{\hat{W}}}} = \underset{|V| \times |V|}{\underline{\underline{C}}} \circ \underset{|V| \times |V|}{\underline{\underline{S}}}}$$

5: • Based on the new affinity measure which quantifies similarity between any pair of images, a new set of ranked lists $\mathcal{T}$ can be obtained. Therefore, the following procedure may be followed:

$$\mathcal{T}^{(0)} \longrightarrow \mathcal{T}^{(1)} \longrightarrow \cdots \longrightarrow \mathcal{T}^{(t)} \longrightarrow \cdots \longrightarrow \mathcal{T}^{(T)}$$

$$\underset{=}{W}^{(0)} \longrightarrow \underset{=}{W}^{(1)} \longrightarrow \longrightarrow \underset{=}{W}^{(t)} \longrightarrow \underset{=}{W}^{(T)}$$

EUCLIDEAN DISTANCE - BASED

• After a given number of iterations we obtain the final set of ranked lists $\mathcal{T}_r$.

## Example

Let $C = \{o_1, o_2, o_3, o_4, o_5, o_6\}$ the set of available images.

Each image $o_j$ is associate with a vertex $v_j$ of the hypergraph

$G = (V, E, w)$ where $V = \{v_1, v_2, v_3, v_4, v_5, v_6\}$.

Let's assume that the initial ranked lists $\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6$ for each image according to the pairwise euclidean distances of their feature vectors are given below:

$$\overleftrightarrow{k=3}$$

$$\tau_1 = (o_1, o_2, o_3, | o_4, o_5, o_6)$$

$$\tau_2 = (o_2, o_3, o_1, | o_5, o_6, o_4)$$

$$\tau_3 = (o_3, o_1, o_4 | o_2, o_5, o_6)$$

$$\tau_4 = (o_4, o_3, o_5, | o_1, o_2, o_6)$$

$$\tau_5 = (o_5, o_6, o_4, | o_3, o_2, o_1)$$

$$\tau_6 = (o_6, o_5, o_4, | o_1, o_2, o_3)$$

Assume that the number of nearest neighbours has been set to $k = 3$, such that:

$$\boxed{w(o_i, o_j) = 1 - \log_k \hat{\tau}_i(o_j)}$$

$$N_3(o_1) = (o_1, o_2, o_3)$$

$$N_3(o_2) = (o_2, o_3, o_1)$$

$$N_3(o_3) = (o_3, o_1, o_4)$$

$$N_3(o_4) = (o_4, o_3, o_5)$$

$$N_3(o_5) = (o_5, o_6, o_4)$$

$$N_3(o_6) = (o_6, o_5, o_4)$$

Thus, the graph could have the following structure: