Confidence-Aware Active Feedback for Interactive Instance Search

Yue Zhang, Chao Liang, and Longxiang Jiang

Abstract—Online relevance feedback (RF) is widely utilized in instance search (INS) tasks to further refine imperfect ranking results, but it often has low interaction efficiency. The active learning (AL) technique addresses this problem by selecting valuable feedback candidates. However, mainstream AL methods require an initial labeled set for a cold start and are often computationally complex to solve. Therefore, they cannot fully satisfy the requirements for online RF in interactive INS tasks. To address this issue, we propose a confidence-aware active feedback method (CAAF) that is specifically designed for online RF in interactive INS tasks. Inspired by the explicit difficulty modeling scheme in self-paced learning, CAAF utilizes a pairwise manifold ranking loss to evaluate the ranking confidence of each unlabeled sample. The ranking confidence improves not only the interaction efficiency by indicating valuable feedback candidates but also the ranking quality by modulating the diffusion weights in manifold ranking. In addition, we design two acceleration strategies, an approximate optimization scheme and a top-K search scheme, to reduce the computational complexity of CAAF. Extensive experiments on both image INS tasks and video INS tasks searching for buildings, landscapes, persons, and human behaviors demonstrate the effectiveness of the proposed method. Notably, in the real-world, large-scale video INS task of NIST TRECVID 2021, CAAF uses 25% fewer feedback samples to achieve a performance that is nearly equivalent to the champion solution. Moreover, with the same number of feedback samples, CAAF’s mAP is 51.9%, significantly surpassing the champion solution by 5.9%. Code is available at https://github.com/nercms-mmap/caaf.

Index Terms—Active learning, Interactive instance search.

I. INTRODUCTION

WITH the explosive development of media technology, image/video instance analysis has received much attention in the past decade [1], [2], [3], [4], [5], [6], [7], [8]. The goal of instance search (INS) is to search for instances of a query from a gallery of images or videos. After the features and initial ranking list are obtained, online relevance feedback (RF) is commonly employed in INS tasks to refine the imperfect retrieval result by asking for extra supervisory information through an interactive process [9], [10]. However, RF often has low interaction efficiency, as not all samples are guaranteed to provide valuable information.

Active learning (AL) [11] addresses this issue by selecting valuable feedback samples that are beneficial for improving the model performance. Particularly, considering the practical difficulty of collecting labeled data and the special requirement of quick response in the online interaction scenario, we need a lightweight AL strategy that does not require a predefined labeled set and that has low computational complexity. However, mainstream AL methods, e.g., [12], [13], [14], [15], require an initial labeled pool that accounts for 5%–10% of the entire training set for a cold start, and solving them is often time-consuming. Therefore, existing AL methods cannot fully satisfy the need for online RF in interactive INS tasks.

To address the above issue, we propose a confidence-aware active feedback method (CAAF) that is specifically designed for online RF in interactive INS tasks. Inspired by the explicit difficulty modeling scheme in self-paced learning (SPL) [16], [17], CAAF measures the confidence of each sample with the manifold ranking (MR) [18] loss. Since MR does not require any labeled samples except for the query, CAAF is compatible with interactive INS tasks where the query is the only labeled sample at the beginning. However, unlike SPL, which focuses on easy samples with small losses, CAAF preferentially selects low-confidence hard samples with large losses, as they have higher potential values from the perspective of AL [19].

To solve the proposed CAAF, we adopt the alternative optimization strategy [20], which divides CAAF into a ranking step and a suggestion step. The ranking step refines ranking scores with the user’s relevance feedback, where labeled samples are endowed with higher confidence scores and have higher weights in diffusing their ranking scores. The suggestion step estimates samples’ confidence scores with their ranking losses, where unlabeled samples with larger ranking losses gain lower confidence scores. As a result, CAAF improves not only the interaction efficiency by selecting valuable feedback samples but also the retrieval performance by modulating the diffusion weights in MR. However, the above solution consists of two quadratic programming problems, which still suffer from high computational complexity. Therefore, we further design two acceleration strategies to reduce the execution time of CAAF. We abandon the convergence condition of the alternative optimization strategy and approximately solve the quadratic-programming-based

Manuscript received 29 March 2022; revised 21 July 2022 and 31 August 2022; accepted 24 October 2022. Date of publication 28 October 2022; date of current version 1 November 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 11903214, 61862015, 61876135, and 62071338, and in part by the Fundamental Research Founds for the Central Universities under Grant 2042022K0001. The Associate Editor coordinating the review of this manuscript and approving it for publication was Dr. Mei-Ling Shyu. (Corresponding author: Chao Liang.)

The authors are with the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan 430072, China, and also with the Hubei Key Laboratory of Multimedia and Network Communication Engineering, Wuhan University, Wuhan 430072, China (e-mail: moozy924@whu.edu.cn; cliang@whu.edu.cn; jianglx@whu.edu.cn). Digital Object Identifier 10.1109/TMM.2022.3217965

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply. See https://www.ieee.org/publications/rights/index.html for more information.
Fig. 1. Workflow of the proposed CAAF. Given an initial ranking list \( r^0 \), where gallery samples are listed in descending order by their initial ranking scores, the pairwise similarity between the top-ranked \( K \) gallery samples and the probe is depicted by a confidence-weighted data manifold. On this basis, CAAF couples the ranking score \( f \) and confidence score \( v \) in one objective function \( E(f, v) \), where \( f \) is solved by minimizing \( E_v(f) \) in the ranking step and \( v \) is solved by minimizing \( E_f(v) \) in the suggestion step. Feedback suggestions are returned to the user, and the feedback scores are used to refine the ranking scores in the next round of feedback. The re-ranked top-\( K \) samples, which are listed in descending order by their refined ranking scores (represented as a light yellow bar), are merged back to the initial ranking list (represented as a light blue bar), and the final ranking list \( r^* \) is generated.

The contributions of this paper are summarized as follows:

- We propose CAAF, an AL strategy that is specifically designed for interactive INS tasks, which improves not only the interaction efficiency by selecting valuable feedback samples but also the retrieval performance by modulating the diffusion weights in manifold ranking.
- We design two acceleration strategies to solve CAAF, including an approximate solution to simplify the solving process and a top-\( K \) search scheme to reduce the problem scale, which ensure a smooth interaction experience in large-scale INS tasks.
- We conduct extensive experiments on both image and video INS tasks searching for buildings, landscapes, persons, and human behaviors, which demonstrate the effectiveness of the proposed method. Notably, in the large-scale video INS task of NIST TRECVID 2021, CAAF uses 25% fewer feedback samples to achieve nearly equivalent performance as the champion solution. Moreover, with the same number of feedback samples, CAAF’s mAP is 51.9%, significantly surpassing the champion solution by 5.9%.

The remainder of this paper is organized as follows: Section II briefly reviews the related work. In Section III, we describe the proposed CAAF method and its solution. Section IV shows the experimental results, and conclusions are given in Section V.
Our method is a post-processing strategy that aims to refine the existing retrieval results through efficient human interactions. The method has been tested to be effective on image- and video-based datasets covering buildings [26], landscapes [27], persons [28], and human behaviors [29].

B. Relevance Feedback

RF is a commonly utilized post-processing technique that refines the existing retrieval results by asking users to provide extra annotations. Generally, RF comprises two steps: sample selection and ranking optimization. Existing methods mainly focus on the second procedure, i.e., optimizing the model with a small amount of user feedback to generate more accurate retrieval results [9], [10]. However, the first procedure is naively implemented by top selection or random selection, which is not guaranteed to select the most valuable feedback candidates, thus leading to low interaction efficiency [30].

To address this problem, some researchers have introduced AL techniques to RF [30]. CAAF follows this rationale, but implements the idea differently. The details are described in the following sections.

C. Active Learning

AL methods aim to minimize the annotation cost by selecting the most valuable feedback samples that are either informative [31], [32], representative [12], [13], or both [15], [33]. The mainstream AL methods require a pre-defined labeled set for a cold start, e.g., VAAL [13] initializes the labeled pool with 10% of the training set, and CoreGCN [15] randomly selects 1,000 samples as the initial labeled set. In addition, these methods are often computationally complex to solve, e.g., VAAL needs to train a variational autoencoder and an adversarial network to select representative unlabeled samples, and CoreGCN estimates the uncertainty of unlabeled samples by training a graph convolutional network. However, considering the practical difficulty of collecting labeled data and the special requirement of quick response in the online interaction scenario, these methods cannot fully satisfy the need for online RF in interactive INS tasks.

CAAF addresses this problem by measuring the samples’ confidence with the manifold ranking [18] loss and preferentially selects low-confidence hard samples with large losses as valuable feedback candidates. Since MR does not require any labeled samples except for the query, CAAF is compatible with interactive INS tasks where the query is the only labeled sample at the beginning.

D. Self-Paced Learning

SPL [16], [17] is a classic learning paradigm that gradually incorporates easy to hard samples into the training processes. The core idea of SPL is to measure the easiness of each sample with a task-specific loss and to preferentially select easy samples with low losses to train the model. The effectiveness of such a learning regime has been validated in various tasks, including fine-grained visual classification [34], cross-modal matching [35] and object detection [36], [37], [38].

Recently, some methods [14], [19] have combined SPL and AL into one framework, e.g., ASPL [19] assigns pseudo labels for high-confidence easy samples selected by SPL and annotates ground-truth labels for low-confidence samples selected by AL; and SPAL [14] integrates SPL and AL into one objective function, so the easiness and representativeness of each sample can be jointly considered for model training.

Different from ASPL and SPAL, CAAF directly uses SPL as an AL strategy and preferentially selects low-confidence hard samples with large losses as valuable candidates for online RF in interactive INS tasks. In addition, CAAF uses SPL as a modulator to adjust the diffusion weights in MR, where labeled samples are endowed with higher confidence scores and thus have higher weights in diffusing their ranking scores. As a result, CAAF improves not only the interaction efficiency but also the ranking accuracy.

III. PROPOSED METHOD

In this section, we present the problem formulation with necessary notations and then introduce the solution and acceleration strategies.

A. Problem Formulation

Given a probe $p$ and a gallery set $\mathcal{G} = \{g_i\}_{i=1}^m$, we merge them into a new image set $\mathcal{X} = \mathcal{G} \cup \{p\} = \{x_i\}_{i=1}^m$, where $m = n + 1$. Next, the pairwise data affinity can be represented as $\mathbf{A} = [\max(a_{ij}, 0)]_{i,j=1}^m \in [0,1]^{m \times m}$, where $a_{ij}$ reflects the similarity between $x_i$ and $x_j$. Our goal is to actively select the most valuable feedback samples for RF to refine the ranking score $\mathbf{f} = [f_i]_{i=1}^m \in [0,1]^{m \times 1}$, where $f_i$ is the ranking score of $x_i$.

**Manifold ranking:** The classic manifold ranking (MR) [18] aims to obtain the ranking score $\mathbf{f}$ by minimizing a pairwise ranking loss function, with a smoothing term to constrain similar samples to have similar ranking scores, and a fitting term to prevent the ranking score $\mathbf{f}$ from deviating too much from the reference ranking score $\mathbf{y} = [0, \ldots, 0, 1]^\top \in \{0,1\}^{m \times 1}$. The pairwise loss function between $x_i$ and $x_j$ is defined as

$$l_{ij} = a_{ij}(f_i - f_j)^2 + \alpha(f_i - y_i)^2 + \alpha(f_j - y_j)^2,$$

where $\alpha \in (0, 1)$ is a parameter to balance the smoothing term and fitting term. To adapt such a framework to our human-in-the-loop setting, we redefine $\mathbf{y} = [y_i]_{i=1}^m$ as

$$y_i = \begin{cases} s_i, & \text{if } x_i \in \Psi, \\ 0, & \text{otherwise}, \end{cases}$$

where $s_i \in \{0, 1\}$ is the feedback score of $x_i$, and $\Psi$ is the set of all annotated samples. $s_i = 1$ if $x_i$ is relevant to the probe, and $s_i = 0$ otherwise. In particular, probe $x_m = p$ can be considered a special annotated sample whose feedback score is $s_m = 1$, $f_i = y_i$ if $x_i \in \Psi$.

**Confidence modeling:** In the context of INS, the confidence of a sample can be defined as the reliability of its ranking score. We denote the confidence score as $\mathbf{v} = [v_i]_{i=1}^m \in [0,1]^{m \times 1}$, where $v_i$ is the confidence score of $x_i$. Apparently, the labeled samples have higher confidence than the unlabeled samples.
Hence, $v_i$ is initialized as

$$v_i = \begin{cases} 1, & \text{if } x_i \in \Psi, \\ 0, & \text{otherwise}. \end{cases}$$

(3)

For unlabeled samples, however, $v_i$ is negatively correlated with the ranking loss. i.e., the higher the ranking loss is, the lower the confidence. And from the perspective of AL, the lower the confidence is, the higher the feedback value. Therefore, $v$ is optimized by minimizing the following loss function:

$$L(f, v) = \frac{1}{m^2} \sum_{i,j} (v_i + v_j)(l_{ij} - \beta),$$

(4)

where $l_{ij}$ is the pairwise manifold ranking loss defined in (1), and $\beta > 0$ is a loss threshold to measure the confidence of each sample. Specifically, when $l_{ij}$ is smaller than $\beta$, minimizing $L(f, v)$ will make both $v_i$ and $v_j$ approach 1; in contrast, when $l_{ij}$ is larger than $\beta$, minimizing $L(f, v)$ will make both $v_i$ and $v_j$ approach 0.

By optimizing (4), $v$ works in two aspects:

- **Feedback generation**: When taking $v$ itself as the optimization objective, it serves as an indicator to generate feedback suggestions based on the ranking confidence of each unlabeled sample.
- **Weight modulation**: When taking $f$ as the optimization objective, $v$ becomes an additional weight in MR to increase the impact of high-confidence samples and to reduce the influence of the low-confidence samples.

Furthermore, to constrain the element value of $v$, a squared norm regularization term $\mathcal{R}(v)$ is added to (4).

$$\mathcal{R}(v) = \frac{\gamma}{m} \|v\|_2^2,$$

(5)

where $\gamma > 0$ is the regularization weight parameter. We obtain the following constraint optimization problem:

$$\min_{f, v} E(f, v) = L(f, v) + \mathcal{R}(v)$$

s.t. $0 \leq f \leq 1, f_i = y_i$ if $x_i \in \Psi$

$$0 \leq v \leq 1, v_i = 1$$

(6)

where 0 and 1 represent a full 0 vector and a full 1 vector, respectively, and $\leq$ denotes the element-wise comparison of $\leq$.

### B. Solution

To solve $f$ and $v$, we adopt the alternative optimization strategy [20] and decompose (6) into a ranking step and suggestion step.

**Ranking step**: In the ranking step, we optimize $f$ with fixed $v$; thus, (6) can be simplified by eliminating the regularization term $\mathcal{R}(v)$ and the threshold $\beta$. The objective function is then computed as

$$E_f(v) = \frac{1}{2} \sum_{i,j} \tilde{v}_{ij} l_{ij},$$

(7)

where $\tilde{v}_{ij} = v_i + v_j$ and the constant coefficient $\frac{1}{m^2}$ in (4) is replaced by $\frac{1}{2}$ to simplify the following transformation. By replacing $l_{ij}$ with (1) and by eliminating terms irrelevant to $f$, (7) can be further rewritten as a confidence-weighted manifold ranking problem

$$E_v(f) = \frac{1}{2} \sum_{i,j} \tilde{v}_{ij} l_{ij},$$

$$= \frac{1}{2} \sum_{i,j} \tilde{v}_{ij} a_{ij} (f_i - f_j)^2 + \sum_{i,j} \alpha \tilde{v}_{ij} (f_i - y_i)^2$$

$$= \frac{1}{2} \sum_{i,j} \tilde{v}_{ij} a_{ij} (f_i^2 + f_j^2 - 2f_i f_j)$$

$$+ \sum_{i,j} \alpha \tilde{v}_{ij} (f_i^2 + y_i^2 - 2f_i y_i)$$

$$= \sum_{i,j} \tilde{a}_{ij} f_i^2 - \sum_{i,j} \tilde{a}_{ij} f_i f_j + \sum_{i,j} \alpha \tilde{v}_{ij} f_i^2$$

$$- 2 \sum_{i,j} \alpha \tilde{v}_{ij} f_i y_i + \sum_{i,j} \alpha \tilde{v}_{ij} y_i^2$$

$$= f^\top D f - f^\top \tilde{A} f + f^\top Q f - 2f^\top Q y + C$$

$$= f^\top (P + Q) f - 2f^\top Q y + C,$$

(8)

where $\tilde{a}_{ij} = \tilde{v}_{ij} a_{ij}$ is the confidence-weighted affinity between $x_i$ and $x_j$; $D$ is a diagonal matrix, where $d_{ii} = \sum_j \tilde{a}_{ij}$; $Q$ is a diagonal matrix, where $q_{ii} = \sum_j \alpha \tilde{v}_{ij}$; $P = D - \tilde{A}$ is the Laplacian matrix of the confidence-weighted graph; and $C = \sum_{i,j} \alpha \tilde{v}_{ij} y_i^2$ is a constant term irrelevant to $f$. Since both $P$ and $Q$ are positive semi-definite matrices, (8) is a standard convex quadratic programming (QP) problem [39].

**Suggestion step**: In the suggestion step, $f$ is fixed while $v$ is optimized. The related objective function can be concisely rewritten in a matrix form as

$$E_f(v) = \frac{\gamma}{m} \|v\|_2^2 + \frac{1}{m} \sum_{i,j} (v_i + v_j)(l_{ij} - \beta)$$

$$= \frac{\gamma}{m} \sum_{i} v_i^2 + \frac{1}{m} \sum_{i} v_i \tilde{l}_i$$

$$= \frac{\gamma}{m} v^\top I v + \frac{2}{m} \tilde{l}^\top v,$$

(9)

where $I$ is an $m$-by-$m$ unit matrix, $\tilde{l} = [\tilde{l}_i]_{i=1}^{m} \in \mathbb{R}^{m \times 1}$, and $\tilde{l}_i = \sum_j (l_{ij} - \beta)$. Similarly, (9) is also a standard QP problem.

After the above decomposition, CAAF can generate one round of refined ranking scores and valuable feedback suggestions by alternatively optimizing the ranking step and suggestion step until convergence.

### C. Acceleration Strategies

Real-time response is in high demand in interactive INS tasks, yet it may be too time-consuming to iteratively solve the QP problems in the ranking step and suggestion step, especially when the problem scale $m$ is extremely large. Therefore, we design two acceleration strategies to efficiently solve the proposed model, including an approximate solution to simplify the original solution in Section III-B, and a top-$K$ search scheme to reduce the problem scale.
Approximate solution: To reduce the execution time, we abandon the convergence condition in the alternative optimization strategy and successively solve the ranking step and suggestion step only once in each round of feedback. In this case, we do not have to strictly follow the constraints in (8) and (9), and both \( f \) and \( v \) can be solved by closed-form expressions.

In the ranking step, \( f \) can be solved by

\[
f_i = \begin{cases} 
\frac{y_i}{f_i - \min(\hat{f})}, & \text{if } x_i \in \Psi, \\
\frac{\max(\hat{f}) - \min(\hat{f})}{\max(\hat{f}) - \min(\hat{f})}, & \text{otherwise},
\end{cases}
\]

(10)

where \( \min(\cdot) \) and \( \max(\cdot) \) denote the minimum and maximum value, respectively, of a vector, and \( \hat{f} \) is solved by

\[
\hat{f} = (P + Q)^{-1}Qy.
\]

(11)

In the suggestion step, \( v \) can be solved by

\[
v = -\frac{i}{\gamma m}.
\]

(12)

Since we are only concerned with the relative magnitude of \( v \) instead of its real value, the constant terms can be further omitted. Hence, \( v \) is solved by

\[
v = -\hat{l},
\]

(13)

where \( \hat{l} = [\hat{l}_i]_{i=1}^n \) and \( \hat{l}_i = \sum_j t_{ij} \).

Since (10) and (13) only involve basic matrix operations, we can use a GPU to accelerate the computational process. The comparative results between the original OP solution and the GPU-accelerated approximate solution will be discussed in Section IV-F.

After successively solving \( f \) and \( v \), the user is asked to provide feedback scores for the \( q \) least confident unlabeled samples. At the beginning of the next round of feedback, we reuse (3) to update \( v \), i.e., unlabeled samples are considered to be low-confidence while labeled samples are regarded as high-confidence. Once the pre-determined maximum round of feedback \( T \) is reached, the ranking score of probe \( f_m \) is removed, and the ranking score is denoted as \( f^* = [f_i]_{i=1}^n \). The overall procedure of the approximate solution is summarized in Algorithm 1.

\begin{algorithm}
\caption{Approximate solution for CAAF.}
\begin{algorithmic}[1]
\State \textbf{Input:} the image set \( X = \mathcal{G} \cup \{p\} \), where \( \mathcal{G} \) and \( p \) denote the gallery set and probe, respectively, the affinity matrix \( A \), the initial labeled image set \( \Psi = \{p\} \), the number of feedback samples per round \( q \), and the maximum round of feedback \( T \).
\State \textbf{Output:} The ranking score \( f^* \).
\For{\text{in } [0, T]} 
\State Set \( y \) and \( v \) by (2) and (3), respectively;
\State Solve \( f \) with fixed \( v \) by (10);
\State Solve \( v \) with fixed \( f \) by (13);
\State Select \( q \) samples with the smallest \( v \) from \( X \setminus \Psi \), and add them to \( \Psi \) with their feedback scores;
\EndFor
\State \Return \( f^* = [f_i]_{i=1}^n \).
\end{algorithmic}
\end{algorithm}

\textbf{Top-}K\textbf{ search scheme:} Since relevant samples tend to be concentrated at the top of the initial ranking list [40], we follow a common practice that selects only the top-\( K \) samples from the initial ranking list \( r^0 \) to form the gallery set \( \mathcal{G} \) [41], [42], i.e., the number of galleries \( n = K \), and the total number of samples \( m = K + 1 \). Therefore, we can set a relatively small \( K \) to balance the performance and computational cost. Eventually, the final ranking list \( r^* \) concatenates two parts: (1) the re-ranked top-\( K \) samples listed in descending order by their refined ranking scores \( f^* \) and (2) the remaining samples in their initial ranking orders. The rationality of this practice will also be discussed in Section IV-F.

IV. EXPERIMENTS

This section is divided into six parts. In the first part, we introduce the settings in our experiments. Next, we report the comparative results with existing AL methods on three image-based INS datasets in the second part. The evaluation results on NIST TRECVID are reported in the third part. In the fourth part, ablation studies are performed to demonstrate the effectiveness of the two key modules in CAAF. In the fifth part, we provide a detailed analysis of the confidence modeling scheme introduced in Section III-A to show how \( v \) works in CAAF. The sixth part analyzes the rationality of the proposed acceleration strategies. We discuss the limitations of CAAF in the last part.

A. SETTINGS

1)Datasets: We evaluate our method on three image retrieval datasets and the INS task of NIST TRECVID\footnote{Online}. Available: https://www-nlpir.nist.gov/projects/tv2021/ins.html

- **Holidays** [27] is one of the most widely employed image retrieval benchmarks. There are 1,491 images of 500 landscapes collected from personal holiday albums, and each landscape has one query.

- **Oxford5k** [26] is also a popular dataset for image retrieval. This dataset contains 5,062 images of 11 buildings in Oxford University, and 5 queries are defined for each building, i.e., there are 55 queries in total.

- **CUHK03** [28] is a person re-identification dataset that collects images of 1,467 pedestrians on the CUHK campus. It provides two types of annotations: the first annotation is obtained by manually labeled bounding boxes (labeled set), and the second annotation is obtained by bounding boxes produced by an automatic detector (detected set). We choose the latter in our experiment, as it is more challenging than the former. We follow the new training/testing protocol introduced by Zheng et al. [43], where there are 1,400 probes vs. 5,332 galleries in the testing set.

- **NIST TRECVID INS** Evaluation asks participants to retrieve all relevant video segments about specific topics from the \textit{EastEnders} series. Each participant can submit up to 4 automatic results and 4 interactive results. When generating the interactive results, human interactions are allowed to refine the automatically generated search results, and the
interaction time for each search topic is limited to 5 minutes. The entire dataset contains more than 470,000 video shots from 244 episodes, with a total length greater than 464 hours. We choose 40 query topics about specific persons doing specific actions from the 2020 and 2021 challenges.

2) Features: We choose the 4,096-dimensional feature extracted by VGG16 [44] pretrained on ImageNet [45] for Holidays and Oxford5k, and the 1,536-dimensional feature extracted by Beyond Part Model [46] is adopted for CUHK03. The affinity matrix $A$ is constructed by cosine similarity, and the initial ranking list $r_i^0$ is generated by sorting all gallery samples in descending order by their Euclidean distance to the probe.

For NIST TRECVID, we concatenate the facial visual feature of the character that appeared in each video segment and the category encoding of action taken by the character as the semantic feature of each video segment [29]. We then take the average feature of the top 25 samples in the initial ranking list as the probe feature, as there is no specific query sample in the INS task of NIST TRECVID. Similar to the three image-based datasets, the affinity matrix $A$ is calculated by the cosine similarity. However, we further add a temporal expansion term to reflect the temporal similarity between each video segment, i.e.,

$$a_{ij} = e^{-\lambda (t_i - t_j)} \cos(x_i, x_j),$$

where $|t_i - t_j|$ calculates the temporal distance between shot $i$ and shot $j$, and $\cos(x_i, x_j)$ denotes the cosine similarity between $x_i$ and $x_j$. The intuition is that temporally consecutive video segments tend to share the same topic, and we set $\lambda = 0.005$. Instead of calculating the Euclidean distance, the initial ranking list on NIST TRECVID is generated by an automatic ranking strategy [47].

3) Evaluation Metrics: The mean average precision (mAP) is mainly used to evaluate the performance on all four datasets. We also measure the execution time taken in each round of feedback to evaluate the execution efficiency.

4) Implementation Details: We set $\alpha = 0.01$ on all datasets to balance the smoothing term and the fitting term in (1). On the three image-based datasets, we set $K = 300$, $q = 5$, and $T = 4$, i.e., the top 300 samples in the initial ranking list are used to refine the retrieval performance, and $q \times T = 20$ feedback samples are generated for each probe in total. On NIST TRECVID, unless otherwise specified, we set $K = 2000$ and $q = 18$, but $T$ is no longer fixed since the total time is fixed in the official evaluation, and we try to take as many feedback rounds as possible. In addition, we replace the binary reference score $y$ with the initial ranking score and set $v = 1$ in the first round of feedback on NIST TRECVID, as the semantic features are somewhat weak to provide adequate information.

B. Comparison With Existing AL Methods

We compare CAAF with 6 AL methods on the three image-based datasets:

- **Rand** is a commonly utilized baseline for AL approaches, where feedback samples are randomly chosen from the unlabeled set.
- **CoreSet** [12] is a state-of-the-art geometric technique and it aims to choose a representative subset of the entire dataset.
- **VAAL** [13] aims to select the most representative samples by training a variational autoencoder and an adversarial network to discriminate between unlabeled data and labeled data. The probability associated with the discriminator’s predictions serves as a score to select low-confidence samples for feedback.
- **SPAL** [14] simultaneously considers the potential value and easiness of an instance by integrating a self-paced regularizer and a term that estimates the distribution difference between labeled data and unlabeled data into one objective function.
- **UncertainGCN** [15] constructs a sequential graph convolutional network (GCN) to distinguish unlabeled samples from labeled samples. The outputs of the GCN serve as confidence scores, and an uncertainty sampling approach is applied to select samples whose confidence scores are closest to a pre-defined small margin.
- **CoreGCN** [15] integrates CoreSet and UncertainGCN to select informative and representative feedback samples.

All baseline methods are implemented with their default settings in our experiment, except for VAAL, as it requires a relatively large number of labeled samples to train the VAE and adversarial network. However, it is impractical for our interactive retrieval settings, where the probe is the only labeled sample at the very beginning. Therefore, we randomly select $q$ samples in the first round of feedback, and VAAL is applied from the second round of feedback. For a fair comparison, all baseline methods, as with CAAF, are applied to the top-$K$ samples in the initial ranking list. In addition, these methods are only used to generate feedback suggestions for RF, and the feedback scores are diffused by the same ranking process described in Section III-B.

Fig. 2 shows the dynamic performance of all AL methods as the feedback round $T$ increases on three datasets. Compared with state-of-the-art AL methods, CAAF gains the highest improvement after 4 rounds of feedback, with the mAP improving more than 15% on Holidays and Oxford5k and more than 45% on CUHK03. Notably, different from the experimental results reported in other literature, some methods, e.g., CoreSet and VAAL, may be inferior to Rand, which can be attributed to the lack of labeled samples. These AL methods usually need a pre-defined labeled set that accounts for 5%~10% of the whole unlabeled set for a cold start, yet the query is the only labeled sample in the first round of feedback in interactive INS tasks. These results demonstrate that CAAF is more compatible with interactive INS tasks compared with state-of-the-art AL methods.

Fig. 3 compares the mean and standard deviation of the execution time in each round of feedback on all three datasets. We observe that most baseline methods need more than 400 ms ($\approx 39$ s) to generate one round of feedback suggestions, which cannot satisfy the optimal computer response time [48]. However, with our approximate solution, CAAF is able to generate feedback suggestions in less than 5 ms, demonstrating its computational efficiency.
Fig. 2. Performance comparison of different AL methods on the Holidays, Oxford5k, and CUHK03 datasets under different feedback rounds (T), where each line represents an AL method.

Fig. 3. Execution time comparison of different AL methods on Holidays, Oxford5k and CUHK03.

C. Evaluation on NIST TRECVID

In this section, we evaluate our method on the INS task of NIST TRECVID 2020 & 2021 to demonstrate the capability of handling extremely large-scale datasets.

1) Evaluation on NIST TRECVID 2021: We report the official evaluation results [29] of the top-3 interactive runs and their corresponding automatic runs in 2021:

- **F_M_A_B_WHU_NERCMS.21_2** (Auto1) is the official first-ranked automatic result.
- **F_M_A_B_WHU_NERCMS.21_4** (Auto2) is the official second-ranked automatic result.
- **I_M_A_B_WHU_NERCMS.21_1** (Auto1+TopK) is the official first-ranked interactive result. This method takes Auto1 as the initial search result, asks the user to verify the initially top-ranked video segments, and removes the irrelevant segments from the search list.
- **I_M_A_B_WHU_NERCMS.21_5** (Auto2+TopK) is the official second-ranked interactive result. This method takes the same interaction strategy as Auto1+TopK, and the only difference is that it takes Auto2, instead of Auto1, as the initial search result.
- **I_M_A_B_WHU_NERCMS.21_3** (Auto2+CAAF) is the official third-ranked interactive result that refines Auto2 via the proposed CAAF.

In addition to the official evaluation results, we supplement an interactive solution (Auto2+CAAF*) where the total feedback number of each topic is equivalent to that of Auto2+TopK, i.e., 176 samples are fed back to the user in Topic 9319, 169 samples are fed back to the user in Topic 9320, etc.

As presented in Table I, when the total interaction time is constrained to 5 minutes, the user can check 105 video segments on average with Auto2+TopK, which is 28 more than...

| Topic  | Auto2+TopK | Auto2+CAAF | Auto2+CAAF* |
|--------|------------|------------|-------------|
|        | #FB        | AP (%)     | #FB         | AP (%)     | #FB         | AP (%)     |
| 9319   | 176        | 59.8       | 75          | 60.0       | 176         | 70.7       |
| 9320   | 169        | 79.0       | 67          | 77.2       | 169         | 80.2       |
| 9321   | 182        | 65.3       | 80          | 64.1       | 182         | 72.1       |
| 9322   | 97         | 56.4       | 60          | 53.2       | 97          | 59.1       |
| 9323   | 90         | 51.5       | 66          | 53.1       | 90          | 57.8       |
| 9324   | 91         | 49.4       | 61          | 52.8       | 91          | 56.0       |
| 9325   | 120        | 71.9       | 77          | 69.7       | 120         | 75.3       |
| 9326   | 138        | 74.5       | 99          | 73.1       | 138         | 78.4       |
| 9327   | 130        | 74.0       | 90          | 74.0       | 130         | 75.7       |
| 9328   | 98         | 20.9       | 86          | 29.0       | 98          | 34.3       |
| 9329   | 114        | 21.5       | 90          | 21.6       | 114         | 24.5       |
| 9330   | 97         | 51.9       | 73          | 51.4       | 97          | 55.2       |
| 9331   | 104        | 39.8       | 80          | 39.9       | 104         | 45.9       |
| 9332   | 116        | 51.6       | 80          | 52.4       | 116         | 60.2       |
| 9333   | 77         | 40.4       | 82          | 38.0       | 77          | 45.9       |
| 9334   | 79         | 35.2       | 83          | 30.6       | 79          | 45.1       |
| 9335   | 60         | 6.5        | 80          | 8.7        | 60          | 14.0       |
| 9336   | 67         | 16.5       | 96          | 15.2       | 67          | 24.0       |
| 9337   | 50         | 12.8       | 72          | 12.9       | 50          | 15.9       |
| 9338   | 53         | 41.8       | 70          | 41.8       | 53          | 46.7       |

| Mean    | 105.4      | 46.0       | 78.4        | 45.9       | 105.4       | 51.9       |
Fig. 4. 11-point interpolated precision-recall curve of Auto2+CAAF* and all baseline methods in NIST TRECVID 2021, where the values in the legend represent the mAP of each method.

Table II

| Method                  | mAP (%) |
|-------------------------|---------|
| AutoBest [49]           | 25.2    |
| AutoBest+InterBest [49] | 36.8    |
| Auto2                   | 24.7    |
| Auto2+TopK              | 38.1    |
| Auto2+CAAF              | 38.7    |

that of Auto2+CAAF. The low-confidence samples selected by CAAF can sometimes be difficult for human users to discriminate. To alleviate the effects caused by noisy annotations, we ask the user to abstain from feedback if he or she is unsure about the label. Nevertheless, Auto2+TopK only performs 0.1% better than Auto2+CAAF with the same initial search result. With both the same initial search result and total feedback number, Auto2+CAAF* outperforms Auto2+TopK by 5.9%. Moreover, Fig. 4 shows that Auto2+CAAF* even surpasses the first-ranked Auto1+TopK, although the performance of Auto2 itself is inferior to that of Auto1.

All the above experimental results demonstrate that CAAF is capable of handling massive data and that CAAF is more efficient than directly checking the top-ranked samples in the initial ranking result.

2) Evaluation on NIST TRECVID 2020: We further implement Auto2+TopK and Auto2+CAAF on NIST TRECVID 2020. We fix the total feedback number of each topic to 80, as CAAF can check approximately 78 samples in 5 minutes according to the statistics in Table I. And we set $K = 2000$, $q = 16$ and $T = 5$ for CAAF. The results are shown in Table II. As a reference, we also report the best automatic and interactive results (denoted as “AutoBest” and “AutoBest+InterBest,” respectively) on the official evaluation in 2020 [49].

We observe that although Auto2 is inferior to AutoBest, Auto2+TopK outperforms AutoBest+InterBest by 1.3%, which means that TopK is quite a strong baseline. However, Auto2+CAAF outperforms Auto2+TopK by 0.6%, which further demonstrates the effectiveness of CAAF.

D. Ablation Studies

In this section, we examine the roles of two key modules, the ranking step and suggestion step, on the Oxford5k dataset by removing them from CAAF. When the ranking step is removed, the ranking scores of the labeled samples are directly replaced by their feedback scores, while the ranking scores of the unlabeled samples retain their initial ranking scores. And when the suggestion step is removed, the feedback suggestions are generated by random sampling.

We observe from Fig. 5 that the performance of CAAF significantly drops once the ranking step or suggestion step is removed, which demonstrates that each module is essential to CAAF. Since the ranking step can be regarded as modified MR, while the suggestion step serves as an AL sampling strategy, it also implies that CAAF effectively takes advantage of both MR and AL.

E. Analysis of Confidence Modeling

We claim in Section III-A that the confidence modeling scheme in CAAF can not only indicate valuable feedback samples for AL but also modulate the propagation weight in MR and improve the ranking accuracy. The comparative results in Section IV-B have already demonstrated the former part of the claim; this section is divided into two parts to further prove the claim. In the first part, we analyze the effect of weight modulation to demonstrate the latter part of the claim. In the second part, we further analyze the distribution of feedback samples and present an illustrative example of the feedback samples. The experiments are conducted on three image-based datasets with default settings.

1) Analysis of Weight Modulation: To eliminate the effect of weight modulation in the ranking step, we set $v = 1$ at the beginning of each round of feedback. In this case, the ranking step of CAAF degrades to the classical MR. We then compare CAAF and MR with the same suggestion step; the results are shown in Table III.

As shown in Table III, although CAAF performs slightly worse than MR when $T = 0$, i.e., no sample is labeled except for the probe, it consistently outperforms MR on all three datasets.

Fig. 5. Ablation study on the Oxford5k dataset.
TABLE III

| T | Method | Holidays mAP (%) | Oxford5k mAP (%) | CUHK03 mAP (%) |
|---|--------|----------------|-----------------|----------------|
| 0 | MR     | 67.80          | 43.35           | 53.72          |
|   | CAAF   | 67.14          | 43.19           | 53.90          |
| 1 | MR     | 79.02          | 49.53           | 71.14          |
|   | CAAF   | 79.83          | 50.75           | 72.17          |
| 2 | MR     | 82.30          | 53.51           | 83.87          |
|   | CAAF   | 84.02          | 55.78           | 85.58          |
| 3 | MR     | 84.83          | 56.44           | 88.25          |
|   | CAAF   | 86.21          | 59.46           | 89.97          |
| 4 | MR     | 86.29          | 58.39           | 90.60          |
|   | CAAF   | 87.46          | 61.51           | 91.33          |

Fig. 7. Performance gain/manifold smoothing loss v.s. query ID on Oxford5k, where the blue line and orange line represent the performance gain and manifold smoothing loss, respectively, and the dashed lines are their second-order polynomial trend lines.

TABLE IV

| Dataset | Method | mAP (%) | time (ms) |
|---------|--------|---------|-----------|
| Holidays| QP     | 87.60   | 50.8 ± 4.00 |
|         | Appr.  | 87.46   | 4.3 ± 0.26  |
| Oxford5k| QP     | 61.43   | 49.6 ± 0.10 |
|         | Appr.  | 61.51   | 4.4 ± 0.22  |
| CUHK03  | QP     | 91.38   | 50.4 ± 2.10 |
|         | Appr.  | 91.33   | 4.0 ± 0.02  |

A visualized example of the feedback samples selected by CAAF is shown in Fig. 10. As T increases, the number of relevant samples gradually decreases, and so does the initial ranking of the selected samples.

F. Analysis of Acceleration Strategies

This section explores the rationality of the two acceleration strategies introduced in Section III-C on the three image-based datasets with the default settings.

1) Analysis of Approximate Solution: We compare the search accuracy and execution time taken in each round of feedback of both the GPU-accelerated approximate solution (Appr.) and the original QP solution (QP); the results are shown in Table IV. We observe that Appr. achieves equivalent performance as QP; however, the former’s execution time is only approximately one-tenth of the latter’s. Therefore, it is rational to simplify the optimization process through the proposed approximation scheme.

2) Analysis of top-K Search Scheme: The search accuracy and execution time with varying Ks are illustrated in Fig. 6. We observe that the performance tends to be stable when K ≥ 300. The execution time per round of query exponentially increases as K increases. These results demonstrate that selecting only the top-K galleries is a reasonable practice for balancing search accuracy and time cost, and we set K = 300.

G. Limitations

When analyzing the detailed performance gain of each query after 4 rounds of interaction with CAAF, we observe some
Fig. 8. Proportion of relevant and irrelevant feedback samples generated by CAAF, where the blue bars and orange bars represent the proportion (%) of relevant samples and irrelevant samples, respectively, in each round of feedback.

Fig. 9. Heatmap of how feedback samples distribute in the initial ranking list, where the numerical values represent the proportion (%) of feedback samples that are located at different initial ranking positions in each round of feedback, and the deeper color reflects the larger proportion.

Fig. 10. Visualized example of the feedback samples selected by CAAF on the Oxford5k dataset, where the first line of each row shows the feedback samples in the $T$-th round, and the second line shows their initial rankings. Images with blue bounding boxes and orange bounding boxes denote relevant samples and irrelevant samples, respectively.
cases where CAAF achieves minimal performance gain and even degradation. Since CAAF is built on the manifold ranking framework, the effectiveness of CAAF is limited when the data distribution does not conform to the manifold assumption.

To validate this argument, we compute the manifold smoothing loss for each query with its truth ground label by \( \frac{1}{m} \sum_{i=1}^{m} \sum_{j=1}^{m} a_{ij} (y_i - y_j)^2 \), where \( a_{ij} \) is the visual similarity between pairwise samples \( x_i \) and \( x_j \), and \( y_i = 1 \) if \( x_i \) is the ground truth and \( y_i = 0 \), otherwise. Next, we rank each query's ID according to its performance gain in ascending order, and plot the corresponding manifold smoothing loss on Oxford5k. The results are shown in Fig. 7, where the blue line and orange line represent the performance gain and manifold smoothing loss, respectively, and the dashed lines denote their second-order polynomial trend lines. The figure shows that queries with higher loss values tend to obtain lower performance gain, which confirms the above argument.

V. CONCLUSION

This paper investigates CAAF, a method that is specifically designed for interactive INS to improve the interaction efficiency by selecting the most valuable samples for RF. The core idea and main novelty lies in the explicit assessment of the ranking confidence, which improves not only the interaction efficiency by indicating valuable feedback candidates but also the retrieval performance by modulating the weights in the ranking loss. Furthermore, with an approximate solution and a top-K search scheme, CAAF can be efficiently applied to interactive INS on large-scale datasets. Extensive experiments on both image INS tasks and video INS tasks demonstrate the effectiveness of our proposed method.

ACKNOWLEDGMENT

Dr. Chao Liang would like to thank Prof. Philip S. Yu for his selfless help and valuable guidance about this work during Dr. Chao Liang’s academic visit to UIC in 2018-2019. The numerical calculations in this article have been done on the supercomputing system in the Supercomputing Center of Wuhan University.

REFERENCES

[1] T. Lin, M. Yang, C. Tsui, and Y. F. Wang, “Query-adaptive multiple instance learning for video instance retrieval,” IEEE Trans. Image Process., vol. 24, no. 4, pp. 1330–1340, Apr. 2015.
[2] J. Meng, J. Yuan, J. Yang, G. Wang, and Y. Tan, “Object instance search in videos via spatio-temporal trajectory discovery,” IEEE Trans. Multimedia, vol. 18, no. 1, pp. 116–127, Jan. 2016.
[3] S. D. Bhattacharjee, J. Yuan, H. Huang, J. Meng, and L. Duan, “Query adaptive multiview object instance search and localization using sketches,” IEEE Trans. Multimedia, vol. 20, no. 10, pp. 2761–2773, Oct. 2018.
[4] J. Yuan, Z. Liu, and Y. Wu, “Discriminative video pattern search for efficient action detection,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 33, no. 9, pp. 1728–1743, Sep. 2011.
[5] Y. Liu, Z. Lu, J. Li, T. Yang, and C. Yao, “Deep image-to-video adaptation and fusion networks for action recognition,” IEEE Trans. Image Process., vol. 29, pp. 3168–3182, 2020.
[6] Y. Liu, K. Wang, G. Li, and L. Lin, “Semantics-aware adaptive knowledge distillation for sensor-to-vision action recognition,” IEEE Trans. Image Process., vol. 30, pp. 5573–5588, 2021.
[7] W. Zhao, Z. Guan, H. Luo, J. Peng, and J. Fan, “Deep multiple instance hashing for fast multi-object image search,” IEEE Trans. Image Process., vol. 30, pp. 7995–8007, 2021.
[8] X. Yao et al., “Adaptive deep metric learning for affective image retrieval and classification,” IEEE Trans. Multimedia, vol. 23, pp. 1640–1653, 2021.
[9] Z. Chen, Z. Xu, Y. Zhang, and X. Gu, “Query-free clothing retrieval via implicit relevance feedback,” IEEE Trans. Multimedia, vol. 20, no. 8, pp. 2126–2137, Aug. 2018.
[10] M. Zhao, J. Liu, Z. Zhang, and J. Fan, “A scalable sub-graph regularization for efficient content based image retrieval with long-term relevance feedback enhancement,” Knowl. Based Syst., vol. 212, 2021, Art. no. 106505.
[11] B. Settles, “Active learning literature survey,” Univ. Wisconsin-Madison Dept. Comput. Sci., Madison, WI, USA, Tech. Rep. 1648, 2009.
[12] O. Sener and S. Savarese, “Active learning for convolutional neural networks: A core-set approach,” in Proc. 6th Int. Conf. Learn. Representations, 2018, pp. 1–13.
[13] S. Sinha, S. Bhrahimi, and T. Darrell, “Variational adversarial active learning,” in Proc. IEEE Int. Conf. Comput. Vis., 2019, pp. 5971–5980.
[14] Y. Tang and S. Huang, “Self-paced active learning: Query the right thing at the right time,” in Proc. 33rd AAAI Conf. Artif. Intell., 2019, pp. 5117–5124.
[15] R. Caramalau, B. Bhattacharjee, and T. Kim, “Sequential graph convolutional network for active learning,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2021, pp. 9583–9592.
[16] M. P. Kumar, B. Packer, and D. Koller, “Self-paced learning for latent variable models,” in Proc. Adv. Neural Inf. Process. Syst., 2010, pp. 1189–1197.
[17] L. Jiang et al., “Self-paced learning with diversity,” in Proc. Adv. Neural Inf. Process. Syst., 2014, pp. 2078–2086.
[18] D. Zhou, J. Weston, A. Gretton, O. Bousquet, and B. Schölkopf, “Ranking on data manifolds,” in Proc. Adv. Neural Inf. Process. Syst., 2003, pp. 169–176.
[19] L. Lin, K. Wang, D. Meng, W. Zuo, and L. Zhang, “Active self-paced learning for cost-effective and progressive face identification,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 7, pp. 17–19, Jan. 2018.
[20] J. C. Bezdek and R. J. Hathaway, “Convergence of alternating optimization,” Neural, Parallel Sci. Comput., vol. 11, no. 4, pp. 351–368, 2003.
[21] L. Zheng, Y. Yang, and Q. Tian, “SIFT meets CNN: A decade survey of instance retrieval,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 40, no. 5, pp. 1224–1244, May 2018.
[22] L. Rossetto et al., “Interactive video retrieval in the age of deep learning - detailed evaluation of VBS 2019,” IEEE Trans. Multimedia, vol. 23, pp. 243–256, 2021.
[23] Y. Li, H. Yao, and C. Xu, “Intra-domain consistency enhancement for unsupervised person re-identification,” IEEE Trans. Multimedia, vol. 24, pp. 415–425, 2022.
[24] P. Peng, J. Ma, J. Zhu, J. Xue, and Q. Tian, “Improving object retrieval quality by integration of similarity propagation and query expansion,” IEEE Trans. Multimedia, vol. 21, no. 3, pp. 760–770, Mar. 2019.
[25] H. Ma et al., “QA4RF: A question answering based framework for pseudo relevance feedback,” IEEE Access, vol. 9, pp. 139303–139314, 2021.
[26] J. Philip, O. Cham, M. Isard, J. Sivic, and A. Zisserman, “Object retrieval with large vocabularies and fast spatial matching,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2007, pp. 1–8.
[27] H. Jégou, M. Douze, and C. Schmid, “Hamming embedding and weak geometric consistency for large scale image search,” in Proc. Eur. Conf. Comput. Vis., 2008, vol. 5302, pp. 304–317.
[28] W. Li, R. Zhao, T. Xiao, and X. Wang, “DeepReID: Deep filter pairing neural network for person re-identification,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2014, pp. 152–159.
[29] Y. Niu et al., “WHU-NERCMS at trecvid2021: Instance search task,” in Proc. TRECVID, 2021, pp. 1–9.
[30] X. Zhao and G. Ding, “Query expansion for object retrieval with active learning using bow and CNN feature,” Multimedia Tools Appl., vol. 76, no. 9, pp. 12131–12147, 2017.
[31] M. Rottmann, K. Kahler, and H. Gottschalk, “Deep Bayesian active semi-supervised learning,” in Proc. 17th Int. Conf. Mach. Learn. Appl., 2018, pp. 158–164.
[32] Z. Liu, J. Wang, S. Gong, D. Tao, and H. Lu, “Deep reinforcement active learning for human-in-the-loop person re-identification,” in Proc. IEEE/CVF Int. Conf. Comput. Vis., 2019, pp. 6121–6130.
[33] S. Ebert, M. Fritz, and B. Schiele, “RAFL: A reinforced active learning formulation for object class recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2012, pp. 3626–3633.
[34] X. Wu, J. Chang, Y. Lai, J. Yang, and Q. Tian, “BiSPL: Bidirectional self-paced learning for recognition from web data,” IEEE Trans. Image Process., vol. 30, pp. 6512–6527, 2021.

[35] J. Wei, X. Xu, Z. Wang, and G. Wang, “Meta self-paced learning for cross-modal matching,” in Proc. 29th ACM Int. Conf. Multimedia, 2021, pp. 3835–3843.

[36] D. Zhang, J. Han, L. Zhao, and D. Meng, “Leveraging prior-knowledge for weakly supervised object detection under a collaborative self-paced curriculum learning framework,” Int. J. Comput. Vis., vol. 127, no. 4, pp. 363–380, 2019.

[37] D. Zhang, J. Han, L. Yang, and D. Xu, “SPFTN: A joint learning framework for localizing and segmenting objects in weakly labeled videos,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 42, no. 2, pp. 475–489, Feb. 2020.

[38] D. Zhang, H. Tian, and J. Han, “Few-cost salient object detection with adversarial-paced learning,” in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 12236–12247.

[39] S. P. Boyd, L. Vandenberghe, S. P. Boyd, and L. Vandenberghe, Convex Optimization. Cambridge, U.K.: Cambridge Univ. Press, 2004.

[40] J. Zobel, “How reliable are the results of large-scale information retrieval experiments?,” in Proc. 21th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 1998, pp. 307–314.

[41] X. He, W. Min, D. Cai, and K. Zhou, “Laplacian optimal design for image retrieval,” in Proc. 30th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval, 2007, pp. 119–126.

[42] C. Han, K. Chen, J. Wang, C. Gao, and N. Sang, “Re-ranking person re-identification with adaptive hard sample mining,” in Proc. Chin. Conf. Pattern Recognit. Comput. Vis., 2018, vol. 11256, pp. 3–14.

[43] Z. Zhong, L. Zheng, D. Cao, and S. Li, “Re-ranking person re-identification with k-reciprocal encoding,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2017, pp. 3652–3661.

[44] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in Proc. 3rd Int. Conf. Learn. Representations, 2015, pp. 1–14.

[45] J. Deng et al., “ImageNet: A large-scale hierarchical image database,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2009, pp. 248–255.

[46] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, “Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline),” in Proc. Eur. Conf. Comput. Vis., 2018, vol. 11208, pp. 501–518.

[47] J. Yang, “Video instance search for composite semantics,” Wuhan University, 2022.

[48] W. J. Doherty and A. J. Thadhani, Laws of UX: Using Psychology to Design Better Products & Services, O’Reilly Media, 2020.

[49] Y. Peng, Z. Ye, J. Zhang, and H. Sun, “Pku_wict at TRECVID 2020: Instance search task,” in Proc. TRECVID, 2020, pp. 1–11.

Yue Zhang received the B.S. degree in computer science and technology from Wuhan University, Wuhan, China, in 2021, where she is currently working toward the M.S. degree. Her research interests include multimedia retrieval, computer vision, and machine learning.

Chao Liang received the Ph.D. degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2012. He is currently an Associate Professor with the National Engineering Research Center for Multimedia Software, School of Computer Science, Wuhan University, Wuhan, China. His research interests include multimedia content analysis and retrieval, computer vision, and pattern recognition. He was the recipient of the Best Paper Award of PCM 2014, and won first place in TRECVID 2021 INS task.

Longxiang Jiang received the B.E. degree in electronic science and technology from the School of Information Engineering, Wuhan University of Technology, Wuhan, China, in 2017, and the M.E. degree in pattern recognition and intelligent system from the School of Computer Science, Wuhan University, Wuhan, China, in 2020. He is currently an Engineer of computer vision and machine learning.