Online Multi-modal Person Search in Videos

Jiangyue Xia[1], Anyi Rao[2⋆], Qingqiu Huang[2], Linning Xu[2], Jiangtao Wen[1], and Dahua Lin[2]

[1] Department of Computer Science and Technology, Tsinghua University
xiajy16@mails.tsinghua.edu.cn, jtwen@tsinghua.edu.cn
[2] CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong
{anyirao, hq016, dhlin}@ie.cuhk.edu.hk, linningxu@link.cuhk.edu.cn

Abstract. The task of searching certain people in videos has seen increasing potential in real-world applications, such as video organization and editing. Most existing approaches are devised to work in an offline manner, where identities can only be inferred after an entire video is examined. This working manner precludes such methods from being applied to online services or those applications that require real-time responses. In this paper, we propose an online person search framework, which can recognize people in a video on the fly. This framework maintains a multi-modal memory bank at its heart as the basis for person recognition, and updates it dynamically with a policy obtained by reinforcement learning. Our experiments on a large movie dataset show that the proposed method is effective, not only achieving remarkable improvements over online schemes but also outperforming offline methods.

Keywords: online person search, multi-modality, dynamic memory bank, uncertain instance cache, reinforcement learning

1 Introduction

Person identification in videos can be specified into different forms and tasks. Among them, *person search with one portrait* is especially related to real-world applications, such as "intelligent fast forwards" on online video platforms and multimedia-oriented web search, and can further benefit video summarization and story understanding. This task is very challenging compared with other person identification problems such as *person Re-ID* [41,8,6] and *person recognition in photo album* [19,43], as the appearance, pose, and clothing of the characters may vary dramatically through the videos. To overcome this difficulty, the research community has explored the use of various modalities [1,5,34,3,11,25,16], such as face, lip motion, body, audio, subtitle, and screenplay.

However, those methods are mainly offline, *i.e.* an instance is compared with the rest to determine its identity, which leads to high computational complexity. Additionally, for scenarios such as suspect discovery in real-time surveillance

⋆ Corresponding author

Fig. 1. Illustration of the memory updating scheme of human movie watching experience. We select out instances of *Elle Woods* in movie *Legally Blonde* (2001) and demonstrate how we update our memory about actress *Reese Witherspoon* with them. The multi-modal memory stores face, body and audio information, which are closely related to human identities

videos and story understanding in live broadcasting, the offline approaches cannot recognize the identities immediately. In this paper, we work on *online* person search to meet the emerging requirement of timely inference.

Online search is very challenging, as decisions need to be made on the fly based on limited memory. The key to this problem is to effectively update the memory so that it can adapt to the changes as the video proceeds. Think about how human tackle with online person search. Suppose we are watching the movie *Legally Blonde* (2001), as shown in Figure 1. When we see the figure of *Elle Woods*, we compare it with previous images stored in our memories to infer the actress's name. There are two possibilities. 1) If the instance appears to be very similar to *Reese Witherspoon*, we recognize her name immediately, and update the impression of *Reese Witherspoon* in our memories with the current looking of *Elle Woods*. Similar processes are also carried out for other cast. When another new instance comes, we continue to compare it with our dynamically updated memory to judge his/her identity. 2) The other possible reaction is that we cannot confirm her identity since her looking is quite different from any cast that exists in our memories. In this case, we stay confused until she appears again and again. We gradually build up our memories on *Elle Woods* and may be capable of recognizing her as *Reese Witherspoon* in the future.

Inspired by this cognitive process, we propose an *online multi-modal searching machine* (OMS). Specifically, to mimic how human recognize characters and store representations in memory, a *dynamic memory bank* is developed to store *face*, *body* and *audio* features of each cast. These multi-modal feature representations are closely related to human identities. The memory bank is dynamically updated to capture the latest changes to the cast's features as new instances come in. To adapt to diverse movie contents and appearance changes, instead of interacting with the memory by a hand-crafted rule, we formulate the process as a decision making problem and design a controller to learn the strategy of memory updating. Motivated by the second case we mentioned above, it is possible

that an instance cannot be recognized as any cast in list at the very beginning, since the initial dynamic memory bank lacks adequate information. We develop an *uncertain instance cache* to keep these temporarily confusing instances for judgments later on. As the online process goes on, more and more instances are recognized and the dynamic memory bank becomes more informative, we select out instances in the cache and make a second decision for them.

Experiments are conducted on *Cast Search in Movies* dataset [16] to verify the effectiveness of our online multi-modal searching method. Thanks to the adaptive multi-modal feature integration and reinforcement learning based memory updating strategy, our approach raises the mAP from 61.24% to 69.08% and outperforms all the online methods. Surprisingly, it achieves better results than offline methods and declines computational cost at the same time.

2 Related Work

Person Identification in Videos. In order to identify characters in videos, frameworks using diverse features have been proposed. What commonly used are visual features of face [1] and body [16,17], audio features of speaking voice [25], text features of subtitle [3,11] and screenplay [5,34], and contextual features of scene and social relation [15]. In [5,34], with the alignment of subtitles and screenplay, time-stamped annotations are acquired to provide supervision of character naming. Nagrani *et al.* [25] train face and voice classifiers in a joint framework to recognize characters. With face and body features, Huang *et al.* [16,17] propagate identity labels through visual and temporal links between the instances. However, most previous studies work on an offline manner, *i.e.* all the instances are compared with each other, and the corresponding identities are inferred after an entire video is examined, which increases computational complexity. In this paper, we propose an online framework that dynamically updates the memory with features of newly identified instances to enable real-time inference. Since text information such as subtitles and screenplay is more difficult to acquire compared with the internal features, we utilize face, body and audio features to infer identities.

Multi-modal Fusion. In person identification methods, fusion of visual and audio features can be classified into two categories: late integration [4,28] and early integration [14,13,30,47]. Late integration methods design a specific classifier for each modality and combine decisions by voting or scoring, while early integration merges features from different modalities by concatenation, weighted summation, or learning joint presentations, etc., before decision. Erzin *et al.* [4] determine the reliable modality combinations with a cascade of classifiers. Hu *et al.* [14,30] propose a cross-modality weight sharing LSTM to capture correlation of face and audio features for speaker identification. In this paper, the strategy of multi-modal fusion is learnt implicitly in the decision making process.

Memory Modelling. To strengthen the ability of conventional neural networks in modelling long-range temporal dependencies, several memory models are pro-

posed. Graves *et al*. [9] design a Neural Turing Machine (NTM) which holds an external memory to interact with the neural networks through attentional reading and writing operations. While NTM focuses on problems of sorting, copying and recall, Memory Networks [39] utilize large long-term static external memory and target to language and reasoning tasks. Sukhbaatar *et al*. [35] extend the model to a continuous form to enable end-to-end training, making it more generally applicable to tasks with less supervision. These memory models have also been modified to different structures [18,33] and adopted in video-related researches such as summarization [7,38], captioning [37], visual question answering [24] and object tracking [40]. In this paper, we utilize a dynamic memory bank to store updated multi-modal features of cast in movies.

Reinforcement Learning. Reinforcement learning (RL) is a technique for solving decision making problems, aiming at learning a policy for the sequence of state-action pairs to obtain maximal rewards [36]. In recent years, RL has been applied in person Re-ID [26,44,20] and face recognition [29]. In [29,26,44], RL is used to find the most representative frames in video sequences, while in [20], RL guides an agent to select informative training samples which are used to finetune a pre-trained Re-ID model. In this paper, we formulate the updating of memory as a decision making problem, where we learn the strategy with RL to maximize recognition accuracy.

3 Online Multi-modal Search

Given the portraits of a list of cast, our goal is to search them in a sequential movie with an online fashion following the human behaviors. To tackle this challenging problem, we propose a novel *online multi-modal searching machine* (OMS) as shown in Figure 2. There are four key components in OMS, *i.e.* multi-modal feature representations (MFR), a dynamic memory bank (DMB), an uncertain instance cache (UIC) and a controller. Each instance is a tracklet and is represented by multi-modal features. It is compared with the cast stored in the memory bank to judge its identity. The controller then determines whether this instance should be used to update memory or put into the uncertain instance cache for later comparisons. The memory bank and the uncertain instance cache are dynamically updated over time, with a strategy operated by the controller. All these components together build an "intelligent machine" to watch a movie and gradually recognize the characters like humans do.

3.1 Multi-modal Feature Representations

When watching a movie, we can identify a person based on various cues, *e.g.* facial appearance, clothing, and even speech. These modalities are complementary to each other. Therefore, it is necessary for us to capture the representations of different modalities for each instance in the movie. Specifically, we take face,

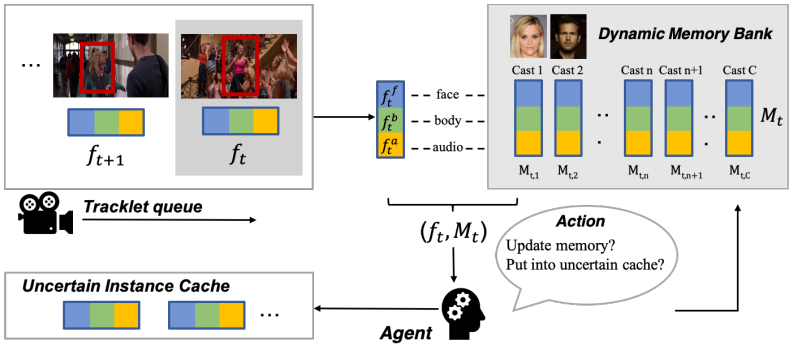

Fig. 2. Pipeline of inference in our proposed OMS. A dynamic memory bank stores the multi-modal feature representations of each actor/actress. When a new instance comes, we compare it with each candidate cast, then the trained agent decides whether to update his memory with this instance or to put it into the uncertain instance cache

body and audio information into consideration in our framework. Given an instance x, we represent it with three feature vectors $(f^f(x), f^b(x), f^a(x))$. Here $f^f \in \mathbb{R}^d$ is the face feature that comes from a face recognition model, $f^b \in \mathbb{R}^d$ is the body feature obtained by a Re-ID network, and $f^a \in \mathbb{R}^d$ is the audio feature acquired by a speech recognition model. These feature vectors are concatenated to form a holistic representation $f = [f^f, f^b, f^a] \in \mathbb{R}^{3d}$.

3.2 Dynamic Memory Bank

A simple way to search cast is to calculate the similarity between the given portrait and the detected instances by their face features. However, as the movie proceeds, the appearance of a cast may change dramatically, and a clear face is missing in many cases where the body is partially occluded or even blurred. Humans can tackle this problem easily with the help of memory. Imagine that when you watch a movie, you may not be able to recognize some of the people at the beginning. However, with the playing of the video, you become more and more familiar with the characters as more identified instances enter the memory.

Inspired by the above observation, we construct a *dynamic memory bank* (DMB) $\mathcal{M}_t \in \mathbb{R}^{C\times 3d}$ to store the most representative features of each person. Here $t \in [1, \cdots, N]$ represents the time when the t-th instance appears, N is the total number of instances in a movie and C denotes the number of cast in list. The memory bank is initialized with the features of the provided portrait of each actor/actress. When an instance x_t comes, we search for it in our memory and then predict its identity. The procedure can be formulated as Eq. 1, where f_t is the multi-modal feature representation of x_t.

$$p_t = \mathcal{M}_t \cdot f_t^T \tag{1}$$

As the movie goes by, the DMB keeps updating, with the strategy shown as Eq. 2. Here $\mu \in [0, 1]$ is a pre-defined updating factor. $\mathcal{G}_{t,j}^1 \in \{0, 1\}$ is a gate of the

controller, the details of which will be introduced in Sec. 3.4, and $j \in [1, \cdots, C]$ represents the j-th cast.

$$\mathcal{M}_{t+1,j} = (1 - \mu\mathcal{G}^1_{t,j})\mathcal{M}_{t,j} + \mu\mathcal{G}^1_{t,j} f_t \tag{2}$$

3.3 Uncertain Instance Cache

At the beginning of a movie, we are not familiar with the characters. Therefore, it may be quite hard for us to recognize some of the tough samples. For example, if a man appears in the first frame of the movie without a visible face, it is impossible for us to identify him at that time. However, as the movie goes on, we begin to know more about the story and the people. We may suddenly recall the uncertain instance before and recognize him with our stronger knowledge.

Motivated by the fact described above, we build a novel module in our machine to store the uncertain instances temporarily, which is named as *uncertain instance cache* (UIC). We denote the cache as $\mathcal{C} \in \mathbb{R}^{k \times 3d}$. k is the size of the cache, which dynamically changes as time goes on. Whether to place an instance x_t into the cache or not is also represented by a gate of the controller, denoted as $\mathcal{G}^2_t \in \{0, 1\}$, which will be introduced in Sec. 3.4. The updating strategy can be formulated as Eq. 3.

$$\mathcal{C}_k = f_t, \quad k \leftarrow k + 1 \qquad \text{if } \ \mathcal{G}^2_t = 1 \tag{3}$$

Whenever the DMB updates, we recall all the instances in the UIC to make new predictions. Specifically, we compare each instance x_i in the cache with the updated memory bank $\mathcal{M}_t$, as shown in Eq. 4. $\mathcal{C}_i$ ($i \in [1, \cdots, k]$) is the multi-modal feature representation of x_i.

$$p_i = \mathcal{M}_t \cdot \mathcal{C}_i^T \tag{4}$$

The p_i here is not the final prediction of the uncertain instance x_i. Whether x_i can be confidently identified and removed from the cache is controlled by the third gate $\mathcal{G}^3_i \in \{0, 1\}$, the details of which will also be introduced in Sec. 3.4.

3.4 Controller

As we mentioned before, there are three gates, *i.e.* $\mathcal{G}^1_{t,j}, \mathcal{G}^2_t, \mathcal{G}^3_i \in \{0, 1\}$, in our framework. The three gates determine "whether to update the memory with instance x_t", "whether to put x_t into the uncertain cache", and "whether to remove x_i from the cache", respectively. In this section, we will provide details on how to construct a controller with all these three gates.

A Manual Controller. A simple way is to design the gates by setting thresholds for the prediction, *i.e.* the similarity. Eq. 5 shows such a manual controller,

where α, β and γ are three pre-defined thresholds. $\mathcal{F}(\Delta t) = \tau \Delta t$ is a regularization function to control the size of the cache. Here Δt is the duration that an instance is stored in the cache and τ is the weight.

$$\begin{cases} \mathcal{G}_{t,j}^1 &= \text{sgn}(p_{t,j} - \alpha) \\ \mathcal{G}_t^2 &= \prod_{j=1}^{C} \text{sgn}(\beta - p_{t,j}) \\ \mathcal{G}_i^3 &= 1 - \prod_{j=1}^{C} \text{sgn}(\gamma - \mathcal{F}(\Delta t) p_{i,j}) \end{cases}, \quad \text{sgn}(x) = \begin{cases} 1, & \text{if } x >= 0, \\ 0, & otherwise \end{cases} \tag{5}$$

A Learnable Controller. Designing the gates according to some manually designed rules will highly reduce the generality. Also, it is hard for us to search for an optimal value of all the hyper parameters. To make our approach more adaptable, we resort to reinforcement learning (RL) to get a learnable controller. RL is characterized by an agent that continuously interacts and learns from the environment through *trial-and-error* games. Its key characteristics include: 1) lack of supervisor, 2) delayed feedback, 3) sequential decisions, and 4) actions affect states, which accord with the peculiarities of our online memory learning setting. Specifically, at each time step, we do not know if updating memory can earn long-term benefits; we observe the instances and make the judgments sequentially; and the updating of our memory will influence future judgments. RL has the potential to find a better policy to replace naive threshold-based strategy. Here, we take $\mathcal{G}_{t,j}^1$ as an example for analysis.

Problem Formulation. The game we teach our agent to play is learning a policy $\mu_\theta(s)$ to decide whether to update the memory bank. For a new instance x_t with feature representation f_t in the sequential movie, we compare it with $\mathcal{M}_t$, and repeat this procedure for each cast $j \in \{1, \cdots, C\}$.

State. State space here is formulated as $\mathcal{S}_t = (\mathcal{M}_t, f_t)$.

Action. Action space here is a one-dimensional discrete space $\{0, 1\}$. If action 1 is taken, we update the memory as Eq. 2.

Reward. Denote the recognition reward at time step t as r_t. If the action is matched with the ground truth label, *i.e.* if x_t is indeed the person j and action 1 is taken, or x_t is not j and action 0 is taken, then the recognition reward at the current time step is $r_t = 1$. Since the effect of the update can only be reflected in future decisions, we define the long-term reward for each action as the cumulative recognition reward in the near future, $R_t = \sum_{m=t}^{t+T} r_m$. We use deep Q-learning network (DQN) to find the improved policy.

The formulation of a learnable $\mathcal{G}_t^2$ is similar to $\mathcal{G}_{t,j}^1$. Note that we do not employ a learnable $\mathcal{G}_i^3$ here. The reason is that $\mathcal{G}_i^3$ is dependent on the samples in the UIC, yet the cache size is quite small and unstable, with which we are not able to train an agent. Through our study, we find that the manual $\mathcal{G}_i^3$ can work well with the other two learned gates. An extensive analysis on the parameters of $\mathcal{G}_i^3$ is provided in the experiment section.

4 Experiments

4.1 Experimental Settings

Data. To validate the effectiveness of our approach, we conduct experiments on the state-of-the-art large-scale *Cast Search in Movies* (CSM) dataset [16]. Extracted from 192 movies, CSM consists of a *query* set that contains the portraits of $1,218$ cast (the actors and actresses) and a *gallery* set that contains $127K$ instances (tracklets). The movies in CSM are split into training, validation and testing sets without any overlap of cast. The training set contains 115 movies with 739 cast and $79K$ instances, while the testing set holds 58 movies with 332 cast and $32K$ instances, and the rest 19 movies are in the validation set.

Evaluation. Given a query with the form of a portrait, our method should present a ranking of all the instances in the gallery to suggest the corresponding possibilities that the instances and the query share a same identity. Therefore, we use *mean Average Precision* (mAP) to evaluate the performance. The training, validation and testing are under the setting of **"per movie"**, *i.e.* given a query, a ranking of instances from only the specific movie will be returned, which is in accordance with real-world applications such as "intelligent fast forwards". Among the 192 movies in CSM, the average size of query and gallery for each movie is 6.4 and 560.5, respectively.

4.2 Implementation Details

Multi-modal Feature Representations. For each instance in CSM, we collect face, body and audio features to facilitate multi-modal person search. The face and body features are extracted for each frame, and averaged to produce the instance-level descriptors. For body feature, we utilize the IDE descriptor [45] extracted by a ResNet-50 [12], which is pre-trained on ImageNet [32] and finetuned on the training set of CSM. We detect face region [42,27] and extract face feature with a ResNet-101 trained on MS-Celeb-1M [10]. NaverNet [2] pre-trained with AVA-ActiveSpeaker dataset [31] is applied on the instances to align the characters with their speech audio, which distinguishes a character's voice with the others' as well as background noises. With the proper setting of sampling rate and Mel-frequency cepstral coefficients (MFCC) [21] to reduce the noises, ultimately, each speaking instance is assigned with an audio feature.

Memory Initialization and Update. Recall that the multi-modal memory bank is $\mathcal{M} = \{M_f, M_b, M_a\}$, where M_f is initialized with the face features extracted from the IMDb portrait of each actor/actress in the movie, and M_b, M_a are void. The optimal μ in Eq. 2 is set to 0.01 through grid search.

RL Training. The DQN mentioned above is instantiated by a two-layer fully-connected network. The training epoch is 100 with learning rate 0.001. Each epoch is run on the whole movie list, with each movie taking 200 Q-learning

Table 1. Person search results on CSM under "per movie" setting

Methods	online	mAP (%, ↑)	complexity * (↓)
Face matching	✓	61.24	$\mathcal{O}(NC)$
TwoStep [22] (face+body)		64.79	$\mathcal{O}(NC)$
TwoStep [22] (face+body+audio)		64.40	$\mathcal{O}(NC)$
LP [46]		9.33	$\mathcal{O}(NC + N^2)$
PPCC [16]		67.99	$\mathcal{O}(NC + N^2)$
OMS (DMB w/ manual updating rule)	✓	63.83	$\mathcal{O}(NC)$
OMS-R (DMB w/ RLC)	✓	64.39	$\mathcal{O}(NC)$
OMS-RM (DMB w/ RLC+MFR)	✓	66.42	$\mathcal{O}(NC)$
OMS-RMQ (DMB w/ RLC+MFR+UIC)	✓	**69.08**	$\mathcal{O}(NC + \hat{k}NC)$

* N: number of instances; C: number of cast; $\hat{k}$: average size of UIC

iterations. The future reward length is set to be 30. We run the framework on a desktop with a TITAN X GPU.

4.3 Quantitative Results

We compare our method with five baselines: **1) Face matching (online):** The instances are sequentially compared with the cast portraits by face feature similarity, without memory updating. **2) TwoStep (face+body):** After comparisons between face features, instances with high recognition confidence are assigned with identity labels, then a round of body feature comparisons is conducted. **3) TwoStep (face+body+audio):** The second step of comparisons in 2) is based on the combination of body and audio features. **4) LP:** The identities of labeled nodes are propagated to the unlabeled nodes with conventional linear diffusion [46] through multi-modal feature links, where a node updates its probability vector by taking a linear combination of vectors from the neighbors [16]. In addition to face features, the body and audio features are combined for matching, where the weights are 0.9 and 0.1, respectively. **5) PPCC** [16]**:** Based on the combination of visual and temporal links, the label propagation scheme only spreads identity information when there is high certainty.

Moreover, four variants of our OMS method are compared to validate the influences of different modules. For *DMB with manual updating rule*, only face features are compared between instances and cast in the memory. When the face similarity exceeds a fixed threshold, the memory is updated with the newly recognized instance. The RL-based controller, multi-modal feature representations and UIC are added sequentially to form the other three variants.

We compare different approaches in three aspects: (1) feasibility of online inference; (2) effectiveness measured by mAP; and (3) computational complexity.

The results are presented in Table 1, from which we can see that: 1) Almost all the previous works tackle this problem in an offline manner except for the simple face matching baseline, while OMS can handle the online scenarios. 2) OMS is quite effective, which can even outperform the offline methods significantly. 3)

The computational cost of OMS is low. Without UIC, OMS is as efficient as the matching-based methods, *e.g.* face matching. Note that the cache size $\hat{k}$ is usually smaller than N and C is less than 10 here. Therefore, even for the complete version of OMS, *i.e.* OMS-RMQ, the complexity is still lower than the popular propagation-based methods [46,16]. 4) The gradually added components of OMS can continuously raise the performances, which proves the effectiveness of the design for each module. All these results demonstrate that OMS is an effective and efficient framework for person search.

4.4 Ablation Studies

What is the behavior of the framework along the time? To discover the behavior of our online framework, we study the development of UIC along the time in our OMS-RMQ method. We select the movies with above 600 instances and record their varied cache sizes at each time step during testing, and average the results among all the movies. The result is presented in Figure 3 (a). Each time step represents that a decision is made to an instance, and the first 600 steps are shown. The "total size" denotes the cumulative number of uncertain instances that have been put into the cache, while "current size" indicates the number of existing instances therein. It is observed that as time goes, total cache size increases gradually. After processing 600 instances, there are around 100 instances that have ever been put into the UIC. The current cache size raises at the very beginning. After around 350 time steps, it drops gradually to zero. It demonstrates that our DMB becomes better after absorbing informative features to assist recognition, thus more and more uncertain instances get a confident result and are popped out of the cache.

Additionally, we record the cumulative recall of instance identification results, namely R@k, along the time. R@k means the fraction of instances where the correct identity is listed within the top k results. The performance improves gradually with time, as shown in Figure 3 (b). The R@1 raises from 59% at the beginning to 67%, which proves the effectiveness of our online design.

What does RL learn? Recall that in the manual rule setting, we update the memory if the similarity between the memory and the instance is higher than a given threshold. With RL, whether to update or not is decided by the trained agent. To have a deeper understanding of how the RL agent makes the decision and why the RL-trained strategy performs better than manual rules, during testing with OMS-RMQ method, we record the similarity scores on different modalities when an instance is used to update the memory. After regressing the data points into Gaussian distributions as shown in Figure 3 (c), the mean and standard deviation of similarity scores on face, body and audio are 0.89, 0.025 (face), 0.64, 0.063 (body), and 0.46, 0.089 (audio), respectively. The RL agent implicitly adjusts the thresholds of updating memory. Interestingly, the mean values are almost the same with the thresholds we carefully designed before that achieve the highest performance in the manual rule setting.

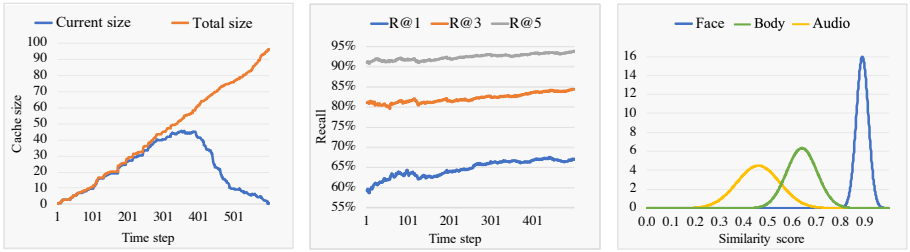

Fig. 3. Ablation studies. (a) The variation of cache size along the time. (b) The development of R@k performance along the time. (c) The distribution of similarity scores of the instances that update the DMB

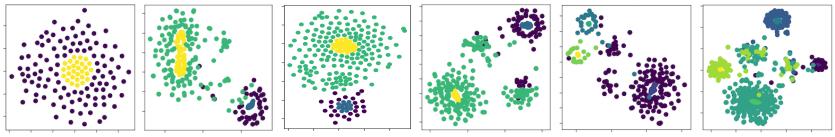

Fig. 4. t-SNE plot of instance features and evolution of memory in 6 movies. Each cluster represents a cast. The "remembered" features in the memory are plotted by light-colored dots, while the instances of a cast are in dark colors. We notice that all the "remembered" features lie at the center of the spread instances. This indicates that the memory absorbs reliable features and well represents the cast's peculiarities

What does memory learn? To prove the effectiveness of the DMB, we visualize the features of a cast's memory and all his/her ground-truth instances in our OMS-RMQ method using t-SNE [23]. Figure 4 shows cast from 6 movies who have at least 15 memory updates, where each cluster represents a cast. We observe that the updated memory features lie at the center of the instance cluster, which indeed provide typical representations of the cast. This shows that our DMB can accurately capture the characteristics of all his/her possible lookings.

How do different modalities work? To study how different modalities contribute to the online multi-modal search, *OMS using DMB with RLC and UIC* is taken as the baseline. The results are shown in Table 2. The performance improves when we gradually add a new modality information in. We observe that the introduction of body and audio features brings 3% and 0.5% improvement to the baseline, respectively. With all these modalities together, OMS achieves a 4.2% enhancement in recognition precision, which validates that all the modalities are complementary to each other and are informative to the online search.

What is the effect of different UIC sizes? As we mentioned above in $\mathcal{G}_i^3$, $\mathcal{F}(\Delta t) = \tau \Delta t$ is the regularization function to control the cache size and τ is the weight. A larger weight leads to a smaller cache, and vice versa. We select differ-

Table 2. Performances of OMS (DMB w/ RLC+UIC) based on different modalities

Method	face	body	audio	mAP (%, ↑)
Face matching (*online*, w/o DMB)	✓			61.24
OMS (DMB w/ RLC+UIC)	✓			64.91
OMS (DMB w/ RLC+UIC)	✓	✓		67.93
OMS (DMB w/ RLC+UIC)	✓		✓	65.39
OMS (DMB w/ RLC+UIC)	✓	✓	✓	**69.08**

Table 3. Performances of OMS (DMB w/ RLC+MFR+UIC) with different cache sizes

weight τ	0	0.04	0.08	0.12	0.16	0.20
mAP (%, ↑)	66.84	68.92	**69.08**	68.13	65.67	63.69
Mean cache size	199	158	96	63	40	16

ent weights and show the corresponding performances provided by OMS-RMQ in Table 3. Under each setting, we record the mean cache size of each movie and average the values among all the movies. The average of mean cache size drops from 199 to 16 as the weight raises from 0 to 0.20. The mAP achieves the maximum 69.08% when the weight is 0.08. When the cache is too small, the uncertain instances are not able to benefit from the gradually absorbed knowledge, which causes inferior performance. Since a character is likely to appear again in the movie before long, a medium-sized cache encourages the uncertain instances to match with a neighboring confidently recognized one. Thus, when the mean cache size is 96, the framework achieves the best result.

4.5 Qualitative Results

Which instances contribute to the memory/are sent into the UIC? In Figure 5, we present some sample instances and the corresponding actions given by the agent during inference. The samples demonstrate that *person search with one portrait* is extremely challenging due to varied illumination, sizes, expressions, poses and clothing. During inference, the trained agent successfully selects informative instances which are mostly easier to recognize to update the memory bank, while the instances that contain profile faces, back figures and occlusions are sent into the UIC for later comparisons when more information is acquired.

Method Comparison. In the "per movie" setting, given a portrait as a query, instances are ranked in descending order according to their similarity to the cast. In Figure 6, we show some searching results provided by our OMS-R and OMS-RMQ methods. The green bounding boxes represent correct recognition, while the red ones are mistakenly identified. It is shown that with the introduction of UIC and multi-modal features, the recognition accuracy is evidently

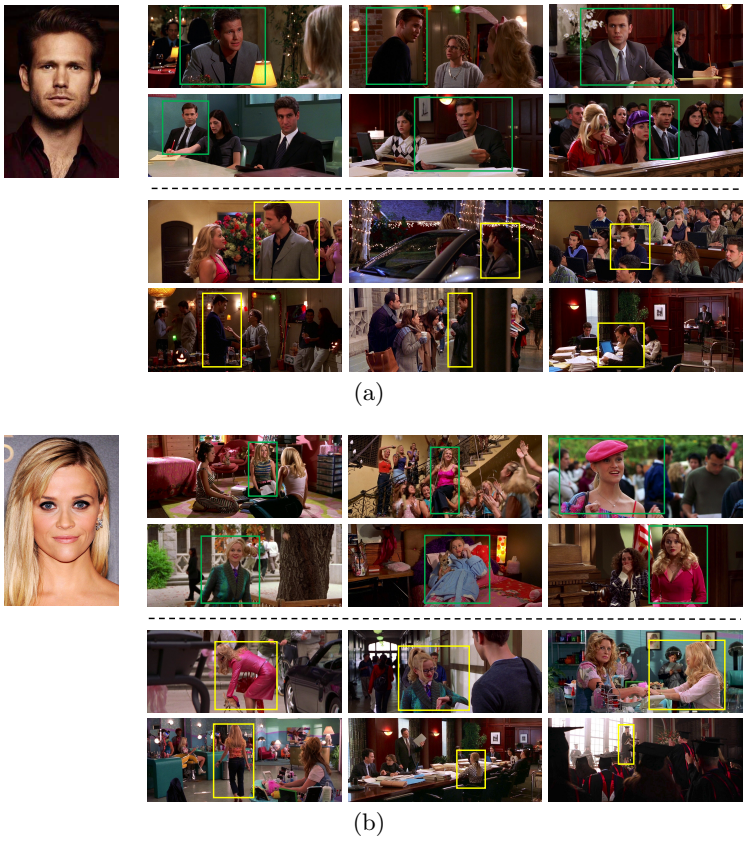

Fig. 5. The sample instances and their corresponding decision making results given by the trained agent. Samples shown above the dash line with green boxes are well-recognized and used to update memory, while those below the dash line with yellow boxes are temporarily put into the UIC. (a) Movie IMDb ID: tt0250494, cast IMDb ID: nm0205127. (b) Movie IMDb ID: tt0250494, cast IMDb ID: nm0000702

improved, which is in accordance with the quantitative result that the mAP raises from 64.39% to 69.08%. Even though the rankings of the samples presented are approaching the length of ground-truth instance list, *i.e.* 11-20/22 and 71-80/109, where instances are harder to recognize due to varied poses and face sizes, OMS-RMQ still provides satisfying results.

5 Conclusion

In this paper, we systematically study the challenging problem of *person search in videos with one portrait*. To meet the demand of timely inference in real-world video-related applications, we propose an *online multi-modal searching machine*. Inspired by the cognitive process in movie watching experience, we construct a

Fig. 6. Samples searched by different methods, ranked in descending order according to similarity. The green bounding boxes represent correct recognition, and the red ones are mistakenly identified. (a) The 11th-20th searching results of the actor's portrait. Movie IMDb ID: tt0072684, cast IMDb ID: nm0578527. (b) The 71th-80th searching results of the actress's portrait. Movie IMDb ID: tt0129387, cast IMDb ID: nm0000139

dynamic memory bank to store multi-modal feature representations of the cast, and develop a controller to determine the strategy of memory updating. An uncertain instance cache is also introduced to temporarily keep unrecognized instances for further comparisons. Experiments show that our method provides remarkable improvements over online schemes and outperforms offline methods.

Acknowledgment: This work is partially supported by the SenseTime Collaborative Grant on Large-scale Multi-modality Analysis (CUHK Agreement No. TS1610626 & No. TS1712093), the General Research Fund (GRF) of Hong Kong (No. 14203518 & No. 14205719), and Innovation and Technology Support Program (ITSP) Tier 2, ITS/431/18F.

References

1. Arandjelovic, O., Zisserman, A.: Automatic face recognition for film character retrieval in feature-length films. In: 2005 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 860–867 (2005) 1, 3
2. Chung, J.S.: Naver at activitynet challenge 2019–task b active speaker detection (ava). arXiv preprint arXiv:1906.10555 (2019) 8
3. Cour, T., Sapp, B., Nagle, A., Taskar, B.: Talking pictures: Temporal grouping and dialog-supervised person recognition. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1014–1021 (2010) 1, 3
4. Erzin, E., Yemez, Y., Tekalp, A.M.: Multimodal speaker identification using an adaptive classifier cascade based on modality reliability. IEEE Transactions on Multimedia **7**(5), 840–852 (2005) 3
5. Everingham, M., Sivic, J., Zisserman, A.: Hello! my name is... buffy" – automatic naming of characters in tv video. In: 2006 British Machine Vision Conference (BMVC). pp. 899–908 (2006) 1, 3
6. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person reidentification by symmetry-driven accumulation of local features. In: 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2360–2367 (2010) 1
7. Feng, L., Li, Z., Kuang, Z., Zhang, W.: Extractive video summarizer with memory augmented neural networks. In: 2018 ACM International Conference on Multimedia (MM). pp. 976–983 (2018) 4
8. Gheissari, N., Sebastian, T.B., Hartley, R.: Person reidentification using spatiotemporal appearance. In: 2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1528–1535 (2006) 1
9. Graves, A., Wayne, G., Danihelka, I.: Neural turing machines. arXiv preprint arXiv:1410.5401 (2014) 4
10. Guo, Y., Zhang, L., Hu, Y., He, X., Gao, J.: Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In: 2016 European Conference on Computer Vision (ECCV). pp. 87–102. Springer (2016) 8
11. Haurilet, M., Tapaswi, M., Al-Halah, Z., Stiefelhagen, R.: Naming tv characters by watching and analyzing dialogs. In: 2016 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1–9 (2016) 1, 3
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016) 8
13. Hu, D., Li, X., Lu, X.: Temporal multimodal learning in audiovisual speech recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3574–3582 (2016) 3
14. Hu, Y., Ren, J.S., Dai, J., Yuan, C., Xu, L., Wang, W.: Deep multimodal speaker naming. In: 2015 ACM International Conference on Multimedia (MM). pp. 1107–1110 (2015) 3
15. Huang, Q., Xiong, Y., Lin, D.: Unifying identification and context learning for person recognition. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2217–2225 (2018) 3
16. Huang, Q., Liu, W., Lin, D.: Person search in videos with one portrait through visual and temporal links. In: 2018 European Conference on Computer Vision (ECCV). pp. 437–454. Springer (2018) 1, 3, 8, 9, 10

17. Huang, Q., Xiong, Y., Rao, A., Wang, J., Lin, D.: Movienet: A holistic dataset for movie understanding. In: 2020 European Conference on Computer Vision (ECCV) (2020) 3
18. Li, D., Kadav, A.: Adaptive memory networks. In: 2018 International Conference on Learning Representations Workshop (ICLRW) (2018) 4
19. Lin, D., Kapoor, A., Hua, G., Baker, S.: Joint people, event, and location recognition in personal photo collections using cross-domain context. In: 2010 European Conference on Computer Vision (ECCV). pp. 243–256. Springer (2010) 1
20. Liu, Z., Wang, J., Gong, S., Lu, H., Tao, D.: Deep reinforcement active learning for human-in-the-loop person re-identification. In: 2019 IEEE International Conference on Computer Vision (ICCV). pp. 6121–6130 (2019) 4
21. Logan, B.: Mel frequency cepstral coefficients for music modeling. In: 2000 International Symposium on Music Information Retrieval (ISMIR) (2000) 8
22. Loy, C.C., Lin, D., Ouyang, W., Xiong, Y., Yang, S., Huang, Q., Zhou, D., Xia, W., Li, Q., Luo, P., et al.: Wider face and pedestrian challenge 2018: Methods and results. arXiv preprint arXiv:1902.06854 (2019) 9
23. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of Machine Learning Research **9**(Nov), 2579–2605 (2008) 11
24. Na, S., Lee, S., Kim, J., Kim, G.: A read-write memory network for movie story understanding. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 677–685 (2017) 4
25. Nagrani, A., Zisserman, A.: From benedict cumberbatch to sherlock holmes: Character identification in tv series without a script. In: 2017 British Machine Vision Conference (BMVC). pp. 107.1–107.13 (2017) 1, 3
26. Ouyang, D., Shao, J., Zhang, Y., Yang, Y., Shen, H.T.: Video-based person re-identification via self-paced learning and deep reinforcement learning framework. In: 2018 ACM International Conference on Multimedia (MM). pp. 1562–1570 (2018) 4
27. Rao, A., Wang, J., Xu, L., Jiang, X., Huang, Q., Zhou, B., Lin, D.: A unified framework for shot type classification based on subject centric lens. In: 2020 European Conference on Computer Vision (ECCV) (2020) 8
28. Rao, A., Xu, L., Xiong, Y., Xu, G., Huang, Q., Zhou, B., Lin, D.: A local-to-global approach to multi-modal movie scene segmentation. In: 2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 10146–10155 (2020) 3
29. Rao, Y., Lu, J., Zhou, J.: Attention-aware deep reinforcement learning for video face recognition. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 3951–3960 (2017) 4
30. Ren, J.S.J., Hu, Y., Tai, Y., Wang, C., Xu, L., Sun, W., Yan, Q.: Look, listen and learn - a multimodal lstm for speaker identification. In: 2016 AAAI Conference on Artificial Intelligence (AAAI). pp. 3581–3587 (2016) 3
31. Roth, J., Chaudhuri, S., Klejch, O., Marvin, R., Gallagher, A., Kaver, L., Ramaswamy, S., Stopczynski, A., Schmid, C., Xi, Z., Pantofaru, C.: Ava-activespeaker: An audio-visual dataset for active speaker detection. arXiv preprint arXiv:1901.01342 (2019) 8
32. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.S., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015) 8
33. Shen, Y., Tan, S., Hosseini, A., Lin, Z., Sordoni, A., Courville, A.C.: Ordered memory. In: Advances in Neural Information Processing Systems. pp. 5037–5048 (2019) 4

34. Sivic, J., Everingham, M., Zisserman, A.: "Who are you?" - learning person specific classifiers from video. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1145–1152 (2009) 1, 3
35. Sukhbaatar, S., Szlam, A., Weston, J., Fergus, R.: End-to-end memory networks. In: Advances in Neural Information Processing Systems. pp. 2440–2448 (2015) 4
36. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction. IEEE Transactions on Neural Networks **9**(5), 1054–1054 (1998) 4
37. Wang, J., Wang, W., Huang, Y., Wang, L., Tan, T.: Hierarchical memory modelling for video captioning. In: 2018 ACM International Conference on Multimedia (MM). pp. 63–71 (2018) 4
38. Wang, J., Wang, W., Wang, Z., Wang, L., Feng, D., Tan, T.: Stacked memory network for video summarization. In: 2019 ACM International Conference on Multimedia (MM). pp. 836–844 (2019) 4
39. Weston, J., Chopra, S., Bordes, A.: Memory networks. In: 2015 International Conference on Learning Representations (ICLR) (2015) 4
40. Yang, T., Chan, A.B.: Learning dynamic memory networks for object tracking. In: 2018 European Conference on Computer Vision (ECCV). pp. 153–169. Springer (2018) 4
41. Zajdel, W., Zivkovic, Z., Krose, B.J.A.: Keeping track of humans: Have i seen this person before? In: 2005 IEEE International Conference on Robotics and Automation (ICRA). pp. 2081–2086 (2005) 1
42. Zhang, K., Zhang, Z., Li, Z., Qiao, Y.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters **23**(10), 1499–1503 (2016) 8
43. Zhang, N., Paluri, M., Taigman, Y., Fergus, R., Bourdev, L.: Beyond frontal faces: Improving person recognition using multiple cues. In: 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4804–4813 (2015) 1
44. Zhang, W., He, X., Lu, W., Qiao, H., Li, Y.: Feature aggregation with reinforcement learning for video-based person re-identification. IEEE Transactions on Neural Networks and Learning Systems **30**(12), 3847–3852 (2019) 4
45. Zheng, L., Bie, Z., Sun, Y., Wang, J., Su, C., Wang, S., Tian, Q.: Mars: A video benchmark for large-scale person re-identification. In: 2016 European Conference on Computer Vision (ECCV). pp. 868–884 (2016) 8
46. Zhou, D., Bousquet, O., Lal, T.N., Weston, J., Schölkopf, B.: Learning with local and global consistency. In: Advances in Neural Information Processing Systems. pp. 321–328 (2003) 9, 10
47. Zhou, H., Liu, Z., Xu, X., Luo, P., Wang, X.: Vision-infused deep audio inpainting. In: 2019 IEEE International Conference on Computer Vision (ICCV). pp. 283–292 (2019) 3