Joint Representative Selection and Feature Learning: A Semi-Supervised Approach

Suchen Wang[1] Jingjing Meng[2] Junsong Yuan[2] Yap-Peng Tan[1]
[1]Nanyang Technological University [2]State University of New York at Buffalo
{wang.sc, eyptan}@ntu.edu.sg, {jmeng2, jsyuan}@buffalo.edu

Abstract

In this paper, we propose a semi-supervised approach for representative selection, which finds a small set of representatives that can well summarize a large data collection. Given labeled source data and big unlabeled target data, we aim to find representatives in the target data, which can not only represent and associate data points belonging to each labeled category, but also discover novel categories in the target data, if any. To leverage labeled source data, we guide representative selection from labeled source to unlabeled target. We propose a joint optimization framework which alternately optimizes (1) representative selection in the target data and (2) discriminative feature learning from both the source and the target for better representative selection. Experiments on image and video datasets demonstrate that our proposed approach not only finds better representatives, but also can discover novel categories in the target data that are not in the source.

1. Introduction

Representative selection aims to find a small subset of data points that can well represent a big data collection. It has attracted much interest in recent years due to the increasing need of analyzing massive visual data and the limited capacity of our computing and storage resources.

Although the problem of representative selection has been well studied in the literature [12, 38, 20, 9, 13, 29, 28, 7], most previous works apply unsupervised approaches. That is, finding a set of items from the given target data without supervision. However, in many applications, we are not only interested in finding representative items (*i.e.*, which data points are exemplars), but also interested in knowing what they are (*i.e.*, recognizing their categories). In other words, although we can find representatives and associate remaining data samples with them based on the similarity, we do not know the exact category of each representative unless labels are provided.

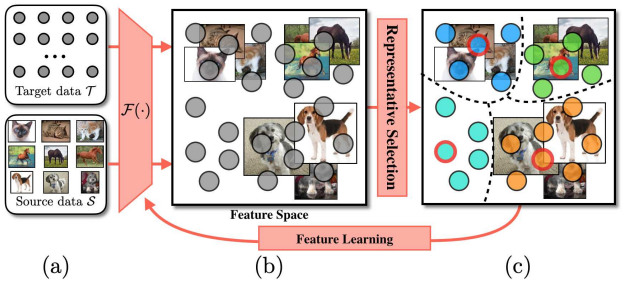


Figure 1. Framework overview. We leverage labeled source data to find representatives from unlabeled target data. Once representatives are found, labels can be naturally transferred from the source to the target. Then we update features for better representative selection. These two steps will alternate until termination.

Recently, Elhamifar *et al*. [8, 7] introduce an additional source set of known items and propose to select source items to represent the target. In this way, target items can be easily recognized by passing category labels of source representatives. Unfortunately, this scheme operates in a closed world assumption, *i.e.*, the source set knows every category that may appear in the target. However, it may not be the case in many real world applications.

In this paper, we take a semi-supervised approach by likewise introducing a source set, but leverage it to find representative items from the target set rather than the source set. More importantly, we do not assume that the source set has covered all categories in the target data. We formulate the representative selection as the facility location problem. We incorporate labeled source data into the objective function such that they can guide the selection for new target data. In the end, connections between the source and the target can be formed via selected representatives, so that we can transfer the labels from the source to the target. As shown in Figure 1(c), we can transfer the labels of "cat", "dog" and "horse" to corresponding groups.

As feature representation plays a critical role in representative selection, we devise a joint optimization framework which alternates between two steps: (1) representative se-

lection and (2) discriminative feature learning. After representatives are found, we leverage both source data and target data to update the features based on their associations with representatives. Subsequently, we reselect representatives and further update the features. This procedure continues until the termination condition is met. The entire process is shown in Figure 1.

The proposed work has the following benefits:

- It leverages the labeled source data to find better representatives for the target.
- The proposed formulation can discover novel categories in the target data.
- The joint representative selection and feature learning can iteratively improve the performance.

Extensive comparisons with state-of-the-arts on two image and two video datasets validate the above benefits.

2. Related Work

In the literature, the problem of representative selection or subset selection has been well studied in many specific applications, such as finding a subset of data to reduce the requirement of computation and memory cost [9, 10, 8], highlighting important shots or events in videos [25, 13, 6, 32, 27, 39, 40, 14] and summarizing a big collection of images [34, 37, 35]. Depending on what information needs to be preserved, the notion of representativeness differs from tasks to tasks. For instance, when selecting a subset of training data to reduce the computations of the training process, the statistical property of data points should be preserved [10, 9]. For the task of the image or video summarization, the diversity and coverage of representatives are taken into account [14, 13, 37, 26].

Representative selection methods may have different objective functions. There are several popular directions in the literature, such as maximum spanning volume [22, 21, 39], sparse coding [10, 25], and facility location [8]. To select a subset of maximum volume, one common way is to apply the determinantal point processes (DPPs) [22]. Recently, many variants of DPPs have been proposed for various applications. For instance, k-DPPs [21] was proposed to handle the fixed number of representatives, Affandi *et al*. [1] combine the DPPs with Markov random field to model the temporal dynamics, and Gong *et al*. [13] propose a learnable scheme for DPPs to select key items from video sequences. For sparse coding [10, 25, 40, 6], the underlying data structures are typically assumed to be linear or subspaces. Elhamifar *et al*. [10] propose to formulate the representative selection as a sparse dictionary selection problem. Meng *et al*. [25] propose to incorporate the locality prior with the dictionary selection to suppress outliers. For facility location [9, 16, 6], data points with the minimum encoding cost (serving cost) are selected as representatives based on the given pairwise similarity or dissimilarity. Its objective is closely related to clustering algorithms [12, 29, 11], which can also be applied for representative selection.

Schemes	Source set	Discover novel class	Label passing
Unsupervised	✗	✓	✗
Semi-supervised [8, 7]	✓	✗	✓
Ours	✓	✓	✓

Table 1. Comparison of three schemes on representative selection.

Although there are various criteria proposed for the subset selection, most of them follow the property of submodularity [35] and, in general, the optimization is NP-hard. To address this issue, many efficient solutions have been studied in the literature. One feasible way is to relax the non-convex objective function to convex and obtain the solution via convex optimization [8]. Another direction is the constant-factor approximation, such as greedy search algorithms [33, 5, 24, 3].

3. Approach

3.1. Problem Statement

Let $\mathcal{T} = \{x_1^t, x_2^t, \ldots, x_n^t\}$ be a target set of n unlabeled items and $\mathcal{S} = \{(x_i^s, y_i^s)\}_{i=1}^m$ be a source set of m labeled items, where each item x_i^s has a corresponding category $y_i^s \in \mathcal{Y}^s$. Our goal is to find a small subset $\mathcal{Z} \subseteq \mathcal{T}$ to well represent the collection of target items, with each representative either represents a known category from $\mathcal{S}$ or a novel category in $\mathcal{T}$. Let $\{z_{ij}^{tt}\}$ be a set of indicators, where $z_{ij}^{tt} \in \{0, 1\}$ denotes the association between x_i^t and x_j^t which is 1 if x_i^t is represented by x_j^t and is 0 otherwise. We also aim to find $\{z_{ij}^{tt}\}$ such that each target item can be represented by a representative of the same category.

In this work, we do not assume that the source set has covered all possible categories and want to discover novel categories in the target set. If the category space of $\mathcal{T}$ is $\mathcal{Y}^t$, we allow $\mathcal{Y}^s \subset \mathcal{Y}^t$. The comparison of our proposed problem and previous problems is summarized in Table 1.

3.2. Preliminary

Representative selection can be formulated as the facility location problem [8]. The objective is to select known source items to represent the target, *i.e*., $\mathcal{Z} \subseteq \mathcal{S}$. However, items of unseen categories may appear in $\mathcal{T}$. In this case, no source item can well represent them. To address this limitation, we reformulate it to find representatives from $\mathcal{T}$ instead of $\mathcal{S}$ by reversing the roles of the source and target data in terms of facilities and clients.

Let $d_{ij}^{st} \in \mathbb{R}$ be the cost of source item x_i^s served by target item x_j^t. We quantify the serving cost as the distance between items in the feature space. Let $\mathcal{F}_\theta(x)$ be the feature

representation of item x, where θ denote all parameters for the representation. We can compute the serving cost d_{ij}^{st} as

$$d_{ij}^{st} = \|\mathcal{F}_\theta(x_i^s) - \mathcal{F}_\theta(x_j^t)\|_2 \tag{1}$$

Let $z_{ij}^{st} \in \{0,1\}$ be a binary indicator of the association between source item x_i^s and target x_j^t, where $z_{ij}^{st} = 1$ if x_i^s is represented by x_j^t and $z_{ij}^{st} = 0$ otherwise. The facility location formulation [8] can be rewritten as

$$\begin{aligned} &\min_{\{z_{ij}^{st}\}} \sum_{i=1}^{m}\sum_{j=1}^{n} z_{ij}^{st} d_{ij}^{st} + \lambda \sum_{j=1}^{n} \mathrm{I}(\|\mathbf{z}_{\cdot j}^{st}\|_p) \\ &\text{subject to} \quad \sum_{j=1}^{n} z_{ij}^{st} = 1, \forall i, \end{aligned} \tag{2}$$

where $\mathbf{z}_{\cdot j}^{st} = [z_{1j}^{st}, \ldots, z_{mj}^{st}]^T$, $\|\cdot\|_p$ is l_p-norm, $\mathrm{I}(\cdot)$ is the indicator function, and λ balances the influence of serving cost (first term) and opening cost (second term). The constraint ensures every source item can be associated with one representative. It is worth noting that $\mathcal{Z}$ can be directly derived from non-zero $\mathbf{z}_{\cdot j}^{st}$, *i.e.*, $\mathcal{Z} = \{x_j^t \in \mathcal{T} \mid \mathbf{z}_{\cdot j}^{st} \neq 0\}$.

In order to well describe target set, $\mathcal{Z}$ should cover all categories in $\mathcal{T}$. However, problem (2) is still limited to existing categories, since it acts to find target items to represent $\mathcal{S}$. Hence, we add target items into the client list such that target items can also be served. Let d_{ij}^{tt} be the cost of target item x_i^t served by x_j^t. We quantify d_{ij}^{tt} as

$$d_{ij}^{tt} = \|\mathcal{F}_\theta(x_i^t) - \mathcal{F}_\theta(x_j^t)\|_2 \tag{3}$$

We can then rewrite the problem as

$$\begin{aligned} &\min_{\{z_{ij}^{st}\},\{z_{ij}^{tt}\}} \sum_{i=1}^{m}\sum_{j=1}^{n} z_{ij}^{st} d_{ij}^{st} + \sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^{tt} d_{ij}^{tt} + \lambda \mathcal{L}_{\text{open}} \\ &\text{subject to} \quad \sum_{j=1}^{n} z_{ij}^{st} = 1, \forall i; \sum_{j=1}^{n} z_{ij}^{tt} = 1, \forall i \end{aligned} \tag{4}$$

Here the opening cost is

$$\mathcal{L}_{\text{open}} = \sum_{j=1}^{n} \mathrm{I}(\| \begin{bmatrix} \mathbf{z}_{\cdot j}^{st} \\ \mathbf{z}_{\cdot j}^{tt} \end{bmatrix} \|_p) \tag{5}$$

where $\mathbf{z}_{\cdot j}^{tt} = [z_{1j}^{tt}, \ldots, z_{nj}^{tt}]^T$. The set of representatives can be derived as $\mathcal{Z} = \{x_j^t \in \mathcal{T} \mid [\mathbf{z}_{\cdot j}^{st}, \mathbf{z}_{\cdot j}^{tt}]^T \neq 0\}$.

Note that the opening loss is actually the cardinality of the selected subset, *i.e.*, $\mathcal{L}_{\text{open}} = |\mathcal{Z}|$. In many applications, the desired number of representatives K could be given in advance. Then the objective of representative selection will turn into minimize the serving cost within the budget. Hence, the problem can be alternatively formulated as

$$\begin{aligned} &\min_{\{z_{ij}^{st}\},\{z_{ij}^{tt}\}} \sum_{i=1}^{m}\sum_{j=1}^{n} z_{ij}^{st} d_{ij}^{st} + \sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^{tt} d_{ij}^{tt} \\ &\text{s.t.} \sum_{j=1}^{n} z_{ij}^{st} = 1, \forall i; \sum_{j=1}^{n} z_{ij}^{tt} = 1, \forall i; \mathcal{L}_{\text{open}} = K \end{aligned} \tag{6}$$

3.3. Joint Representative Selection and Feature Learning

3.3.1 Representative Selection

Initially, we reformulate representative selection as problem (6). Although our primary objective is to find $\{z_{ij}^{tt}\}$, the optimization of $\{z_{ij}^{st}\}$ actually benefits the representative selection for the target data. On one hand, source items can provide prior knowledge about known categories. On the other hand, labels can be transferred from the source to the target based on $\{z_{ij}^{st}\}$. However, problem (6) does not fully exploit the given source labels. Here we incorporate the source labels into the objective function. As we will show in the experiments, the label information can further improve the selection. Suppose the source set $\mathcal{S}$ consists of c categories. We first partition $\mathcal{S}$ into c groups based on their categories, *i.e.*, $\mathcal{S} = \{\mathcal{C}_k\}_{k=1}^{c}$, where $\mathcal{C}_k = \{x_i^s \in \mathcal{S} \mid y_i^s = k\}$. Then we find one target item to represent one entire source group. Let d_{kj}^{ct} be the cost of group $\mathcal{C}_k$ served by x_j^t. We can compute d_{kj}^{ct} by

$$d_{kj}^{ct} = \sum_{i \in \mathcal{C}_k} d_{ij}^{st} = \sum_{i \in \mathcal{C}_k} \|\mathcal{F}_\theta(x_i^s) - \mathcal{F}_\theta(x_j^t)\|_2 \tag{7}$$

Let $z_{kj}^{ct} \in \{0,1\}$ be the association between $\mathcal{C}_k$ and x_j^t, where $z_{kj}^{ct} = 1$ if group $\mathcal{C}_k$ is represented by x_j^t and $z_{kj}^{ct} = 0$ otherwise. Notice that different source groups are in different categories. Basically, no target item can represent two source categories. To avoid this, we add a constraint $\sum_{k=1}^{c} z_{kj}^{ct} \leq 1$, such that each representative can only represent at most one source group. It is to be noted that this constraint plays a crucial role in the feature learning process (in Section 3.3.2). If two source groups are allowed to be represented by one representative, the feature learning will treat them as one category and then mix them up. Then we can rewrite the problem as

$$\begin{aligned} &\min_{\{z_{kj}^{ct}\},\{z_{ij}^{tt}\}} \sum_{k=1}^{c}\sum_{j=1}^{n} z_{kj}^{ct} d_{kj}^{ct} + \sum_{i=1}^{n}\sum_{j=1}^{n} z_{ij}^{tt} d_{ij}^{tt} \\ &\text{subject to} \sum_{j=1}^{n} z_{kj}^{ct} = 1, \forall k; \sum_{j=1}^{n} z_{ij}^{tt} = 1, \forall i; \\ &\sum_{k=1}^{c} z_{kj}^{ct} \leq 1, \forall j; \mathcal{L}_{open} = K, \end{aligned} \tag{8}$$

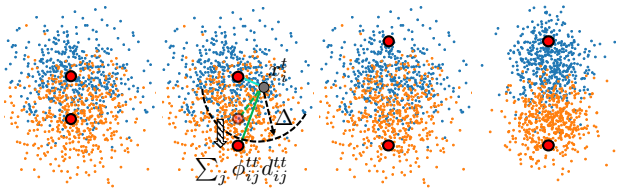


Figure 2. Illustration of the discriminative term. (a) Center points. (b) Discriminative term. (c) Discriminative points. (d) Updated feature. Representatives are highlighted by the red circles. Compared with the representatives in (a), the representatives in (c) are more likely to differentiate two categories, and the neighbors of representatives are more likely to be in the same category.

The opening cost is

$$\mathcal{L}_{\text{open}} = \sum_{j=1}^{n} \mathrm{I}(\| \begin{bmatrix} \mathbf{z}_{\cdot j}^{ct} \\ \mathbf{z}_{\cdot j}^{tt} \end{bmatrix} \|_p) \tag{9}$$

where $\mathbf{z}_{\cdot j}^{ct} = [z_{1j}^{ct}, \ldots, z_{cj}^{ct}]^T$.

In general, facility location formulation tends to select center items in the clusters. However, cluster centers are not the best choice when items of two categories are mixed up. Compared with center points, there are two advantages to select discriminative points as representatives (as shown in Figure 2(a) and 2(c)). First, the discriminative points are more likely to differentiate two categories. Second, the neighbors of discriminative points are more likely to belong to the same category. As we will show in section 3.3.2, it is crucial for the neighbors to be in the same category as the representative, since we will use them to update the feature representation.

In order to promote the discriminative points when clusters are not well separated, we include a discriminative term. To give an intuitive understanding, we illustrate how it works in Figure 2(b). Specifically, let $\phi_{ij}^{tt} \in \{0, 1\}$ be a binary indicator, where $\phi_{ij}^{tt} = 1$ if x_j^t is the second nearest representative of x_i^t and $\phi_{ij}^{tt} = 0$ otherwise. For a target item x_i^t, the distance to its second nearest representative can be expressed as $\sum_j \phi_{ij}^{tt} d_{ij}^{tt}$. We expect this distance can be greater than a margin Δ. Otherwise, there will be a penalty. Likewise, let $\phi_{kj}^{ct} \in \{0, 1\}$ indicate if x_j^t is the second nearest representative of $\mathcal{C}_k$. Our designed discriminative term for the target data and source data can be written as

$$\begin{aligned} \sum_{i=1}^{n}[\Delta - \sum_{j=1}^{n} \phi_{ij}^{tt} d_{ij}^{tt}]_+ &= \sum_{i=1}^{n}\sum_{j=1}^{n} \phi_{ij}^{tt}[\Delta - d_{ij}^{tt}]_+ \\ \sum_{k=1}^{c}[\Delta_k - \sum_{j=1}^{n} \phi_{kj}^{ct} d_{kj}^{ct}]_+ &= \sum_{k=1}^{c}\sum_{j=1}^{n} \phi_{kj}^{ct}[\Delta_k - d_{kj}^{ct}]_+ \end{aligned} \tag{10}$$

where $\Delta_k = |\mathcal{C}_k|\Delta$ and $[\cdot]_+$ is the hinge function. Then our final formulation with the discriminative terms becomes

$$\begin{aligned} \min_{\{z_{kj}^{ct}\},\{z_{ij}^{tt}\}} & \sum_{k=1}^{c}\sum_{j=1}^{n}(z_{kj}^{ct} d_{kj}^{ct} + \phi_{kj}^{ct}[\Delta_k - d_{kj}^{ct}]_+) \\ & + \sum_{i=1}^{n}\sum_{j=1}^{n}(z_{ij}^{tt} d_{ij}^{tt} + \phi_{ij}^{tt}[\Delta - d_{ij}^{tt}]_+) \\ \text{subject to } & \sum_{j=1}^{n} z_{kj}^{ct} = 1, \forall k; \sum_{j=1}^{n} z_{ij}^{tt} = 1, \forall i; \\ & \sum_{k=1}^{c} z_{kj}^{ct} \le 1, \forall j; \mathcal{L}_{\text{open}} = K, \end{aligned} \tag{11}$$

Basically, when clusters are mixed up, discriminative terms penalize center points and promote discriminative points. But if clusters are well separated, the discriminative terms will be zero. Then problem (11) will degrade to problem (8).

3.3.2 Discriminative Feature Learning

In previous sections, we fix θ to optimize the representative selection. Here we will optimize the feature representation $\mathcal{F}_\theta(\cdot)$ based on the built associations $\{z_{kj}^{ct}\}$ and $\{z_{ij}^{tt}\}$.

Most previous representative selection approaches apply a two-stage strategy, *i.e.*, first extract the feature of data points and then find representatives. Before extracting the feature, labeled source data can be used to fine-tune θ for better representation. However, this two-stage strategy is not optimal since the feature learning process is independent of the target data which may consist of novel categories.

To take the target data into consideration, we devise a framework which can alternately find representatives and optimize the representation. By doing this, we can utilize target items to update the representation, including items of novel categories.

We optimize the feature representation by minimizing the triplet loss [31]. Specifically, we reduce the intra-class distances and enlarge the inter-class distances (see Figure 2(d)) so that better $\{z_{ij}^{tt}\}$ can be found in the next selection. Suppose $\{(x_i^a, x_i^p, x_i^n)\}$ is a set of training triplets. We update θ by minimizing

$$\sum_i \left[\Delta + \|\mathcal{F}_\theta(x_i^a) - \mathcal{F}_\theta(x_i^p)\|_2 - \|\mathcal{F}_\theta(x_i^a) - \mathcal{F}_\theta(x_i^n)\|_2\right]_+ \tag{12}$$

The key step is how to construct training triplets. Specifically, the training set consists of three parts. The first part is based on $\mathcal{S}$ by using their labels. We use the same techniques as [31] to create training triplets.

The second part is constructed based on $\{z_{kj}^{ct}\}$ and $\{\phi_{kj}^{ct}\}$ For each source item, we see it as the anchor x_i^a and see its assigned representative as the positive x_i^p. Because the

Algorithm 1: Local Search for Solving Problem (11)

Input : $\mathcal{S}, \mathcal{T}, K$
Output: $\mathcal{Z}$, $\{z_{ij}^{tt}\}$, $\{z_{kj}^{ct}\}$

Initialize $\mathcal{Z}$ *by an arbitrary solution with* $|\mathcal{Z}| = K$;
repeat
  for $x_i^t \in \mathcal{Z}$ **do**
    find $cost(\mathcal{Z})$, $\{z_{ij}^{tt}\}$, $\{z_{kj}^{ct}\}$;
    for $x_j^t \in \mathcal{T} \setminus \mathcal{Z}$ **do**
      $\mathcal{Z}_j = \mathcal{Z} \setminus \{x_i^t\} \cup \{x_j^t\}$;
      find $cost(\mathcal{Z}_j)$;
    j^* = $\arg\max_j$ $cost(\mathcal{Z})$ - $cost(\mathcal{Z}_j)$;
    if $cost(\mathcal{Z}_{j^*}) < cost(\mathcal{Z})$ **then**
      $\mathcal{Z} \leftarrow \mathcal{Z}_{j^*}$;
until *convergence*;

second best representative is from a different group, we see it as the negative x_i^n. For this part, anchors are source items, while positives and negatives are both target items.

The third part is constructed based on $\{z_{ij}^{tt}\}$ and $\{\phi_{ij}^{tt}\}$. Anchors, positives and negatives are all unlabeled target items. To obtain reliable training triplets, we only use target items near the representatives since they are more likely to belong to the same category as the representatives. It is to be noted that the discriminative term is necessary since it can provide more reliable triplets for feature learning. In this part, we treat target items as the anchor x_i^a, associated representatives as the positive x_i^p, and second nearest representative as the negative x_i^n.

3.3.3 Optimization

Here we present the algorithm to solve problem (11) and the entire joint optimization.

In general, the optimization problem (11) is NP-hard [5]. In order to solve it efficiently, we adopt the local search algorithm [17, 3], which is similar to the PAM algorithm [16] in clustering. Let $cost(\cdot)$ denote the serving cost in the problem (11). The algorithm begins with an arbitrary feasible subset with K items. Given the initial subset $\mathcal{Z}$, we need to find the optimal $\{z_{kj}^{ct}\}$ and $\{z_{ij}^{tt}\}$ such that the serving cost is minimal. The optimization of $\{z_{kj}^{ct}\}$ is equivalent to the minimum weight matching problem in the bipartite graph, which can be solved by the Hungarian algorithm [19]. The optimization of $\{z_{ij}^{tt}\}$ can be solved by finding the nearest representative. Then for each current representative, we find a new candidate to replace it such that the decrease of serving cost is the largest. If no candidate can reduce the cost, the algorithm will be terminated. We summarize the local search algorithm in Algorithm 1. The proposed algorithm guarantees convergence since the serving cost is monotonically reduced.

In the feature learning step, the number of triplets whose loss is greater than 0, is a direct clue to terminate our joint optimization. We call those triplets as hard triplets. When there are few hard triplets, further feature learning will not bring better performance but cause oscillation (as shown in Figure 6(b)). We denote τ as the minimal number of hard samples controlling the termination. We will terminate the joint optimization if the number of hard triplets is less than τ or the maximal training epoch is reached. The entire framework can be summarized in Algorithm 2.

Algorithm 2: Joint Optimization

Input : $\mathcal{S}, \mathcal{T}, K, \tau$
Output: $\mathcal{Z}$, $\{z_{ij}^{tt}\}$, $\{z_{kj}^{ct}\}$

repeat
  find $\mathcal{Z}$, $\{z_{ij}^{tt}\}$, $\{z_{kj}^{ct}\}$ by using Algorithm 1;
  create training triplets based on $\mathcal{S}$, $\{z_{kj}^{ct}\}$, $\{z_{ij}^{tt}\}$;
  update θ by minimizing loss (12)
until # *hard triplets* $< \tau$ *or reach maximal epoch*;

4. Experiments

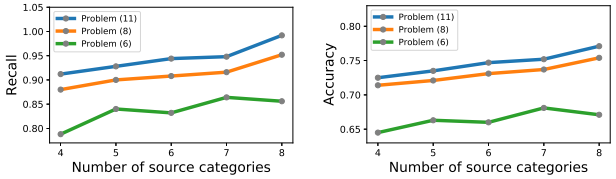
In this section, we evaluate the performance of our proposed method on the task of representative selection. We first conduct the ablation studies to thoroughly investigate each proposed component. For specific applications, we focus on finding key actions from video sequences.

Baselines: We compare our proposed method with the K-medoids clustering (KM) [16], the fixed-size determinantal point processes (kDPP) [21] and two subspace-based methods, sparse modeling representative selection (SMRS) [10] and locally linear reconstruction induced sparse dictionary selection (LLR-SDS) [25]. We also compare the preliminary problem (6) (DS3A) which is modified based on the dissimilarity-based sparse subset selection [8]. For those baselines, we follow the two-stage strategy. That is, we first use the labeled source data to learn the feature representation with the cross-entropy loss (CE) and triplet loss (T) respectively. After the learning process, we apply those approaches to find representatives.

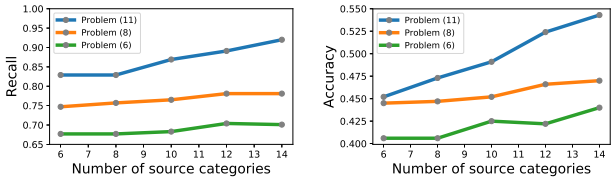
Evaluation Metrics: To evaluate the performance, we consider two factors: (1) how many categories in the target are found by the representatives; (2) whether data points are accurately represented by the representatives belonging to the same category. Correspondingly, we calculate the **recall** of categories in $\mathcal{Z}$ and the **accuracy** of the associations between representatives and target items. If there are n_c out of n target items are correctly associated to its category, the accuracy is computed by n_c/n.

4.1. Ablation Studies

Here we perform some proof-of-concept experiments. The experiments are conducted on two image datasets, **MNIST** [23] and **SCENE15** [30].

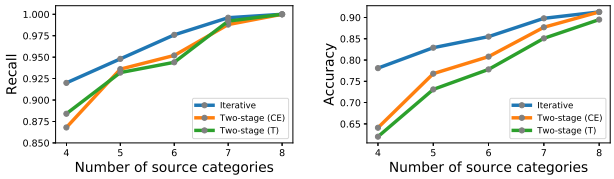


(a) MNIST, Recall (b) MNIST, Accuracy

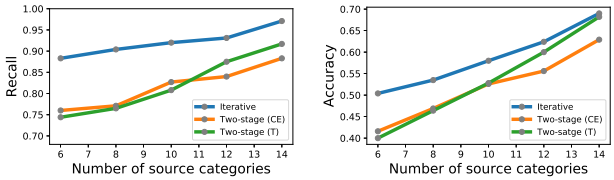


(c) SCENE15, Recall (d) SCENE15, Accuracy

Figure 3. Experimental results of our reformulations on the selection with varying number of source categories.



(a) MNIST, Recall (b) MNIST, Accuracy



(c) SCENE15, Recall (d) SCENE15, Accuracy

Figure 4. Experimental results of different optimization strategies.

	Problem (6)	Problem (8)	Problem (11)
Source labels	✗	✓	✓
Discriminative term	✗	✗	✓

Table 2. Comparison of three proposed problems.

For a fair comparison, we run 25 different experiments and report the averaged results. At each run, we construct the source and target set by random sampling. Specifically, we build the target set with 2,000 randomly selected data points and all categories are included in it (*i.e.*, 10 for MNIST and 15 for SCENE15). For the source set, we randomly select 100 data samples per category and vary the number of categories c' within $\{4, 5, 6, 7, 8\}$ for MNIST and $\{6, 8, 10, 12, 14\}$ for SCENE15. Here we mainly investigate the performance with respect to source categories, the number of representatives is simply set to 10 for MNIST and 15 for SCENE15. We compute the feature of data points from scattering convolution network [4] for MNIST and Resnet18 [15] (pre-trained on ImageNet) for SCENE. We stack two more fully-connected layers and one ℓ_2 normalization layer on the top to learn a new feature representation. We freeze all previous layers and only update newly added layers. Stochastic gradient descent is used to optimize the network. The learning rate and momentum are set to 0.001 and 0.9 respectively. The margin Δ in problem (11) and triplet loss is set to 0.5. The parameter τ is set to 64 and the maximum epoch is 32.

4.1.1 Effectiveness of Proposed Reformulation

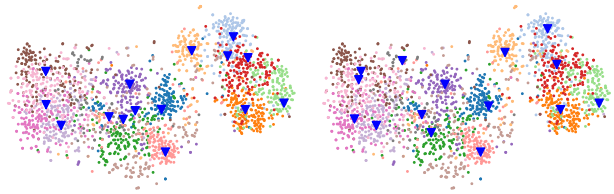
We first evaluate our proposed reformulations on the selection stage. Here no feature learning procedure is performed. We use the feature extracted from the original pre-trained network.

Figure 3 shows the performance of problem (6), (8) and (11). In Table 2, we summarize their differences. All problems are solved by the local search algorithm. From the plots, we have two observations. **(a)** By comparing problem (8) with problem (6), we can see an remarkable improvement as we incorporate the source labels into the objective function. **(b)** By comparing problem (11) with problem (8), we can see that the discriminative terms can offer further performance gains.

Qualitative results Figure 5 visualizes one example of selected representatives on the SCENE15 dataset. The source set includes 14 categories. Problem (11) successfully finds all target categories and achieves a higher accuracy, while problem (8) misses 3 categories. It shows the discriminative term can help to cover more categories and achieve higher accuracy when the feature is not discriminative.

4.1.2 Iterative Optimization Strategy

In this section, we examine the effectiveness of our proposed iterative optimization strategy on feature learning.

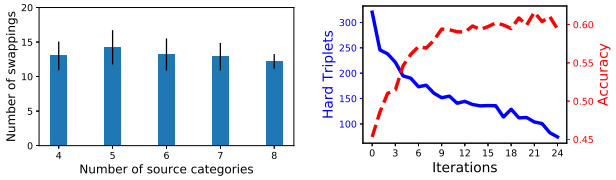


(a) Recall= 0.8, Accr.= 0.453 (b) Recall= 1.0, Accr.= 0.560

Figure 5. T-SNE visualization [36] of selected representatives (highlighted by blue triangles) on SCENE15 dataset. (a) Problem (8) misses 3 categories in the target. (b) Problem (11) is able to cover all categories.

Recall / Accr.	CE+KM	CE+kDPPs	CE+SMRS	CE+LLRSDS	CE + DS3A	T+KM	T+kDPPs	T+SMRS	T+LLRSDS	T+DS3A	Ours
changing tire	0.625 / 0.581	0.567 / 0.529	0.455 / 0.452	0.467 / 0.460	0.613 / 0.592	0.670 / 0.671	0.567 / 0.570	0.651 / 0.595	0.656 / 0.605	0.686 / 0.669	**0.716 / 0.729**
coffee	0.573 / 0.749	0.527 / 0.715	0.453 / 0.642	0.445 / 0.611	0.559 / 0.715	0.638 / 0.806	0.523 / 0.755	0.596 / 0.755	0.622 / 0.747	0.626 / 0.817	**0.663 / 0.842**
cpr	0.700 / 0.699	0.638 / 0.669	0.583 / 0.606	0.555 / 0.541	0.672 / 0.673	**0.773** / 0.777	0.623 / 0.702	0.722 / 0.700	0.736 / 0.682	0.733 / 0.760	0.771 / **0.806**
jump car	0.441 / 0.798	0.419 / 0.771	0.283 / 0.725	0.296 / 0.705	0.451 / 0.774	0.481 / 0.830	0.419 / 0.802	0.487 / 0.800	0.491 / 0.801	0.421 / 0.817	**0.523 / 0.833**
repot	0.570 / 0.653	0.550 / 0.618	0.437 / 0.578	0.427 / 0.571	0.565 / 0.620	0.624 / 0.712	0.579 / 0.658	0.593 / 0.667	0.588 / 0.666	0.580 / 0.722	**0.640 / 0.762**
Average	0.582 / 0.696	0.540 / 0.660	0.442 / 0.601	0.438 / 0.578	0.572 / 0.675	0.637 / 0.759	0.542 / 0.697	0.610 / 0.703	0.619 / 0.700	0.609 / 0.757	**0.663 / 0.794**

Table 3. Experimental results on Narrated instructional dataset with 2.5c representatives. c is the number of categories in the source set.



(a) (b)

Figure 6. (a) The average number of swappings of the local search algorithm. (b) The number of hard triplets and accuracy during the joint optimization (best viewed in color).

We compare it with the conventional two-stage strategy. Both are applied to problem (11) to find representatives.

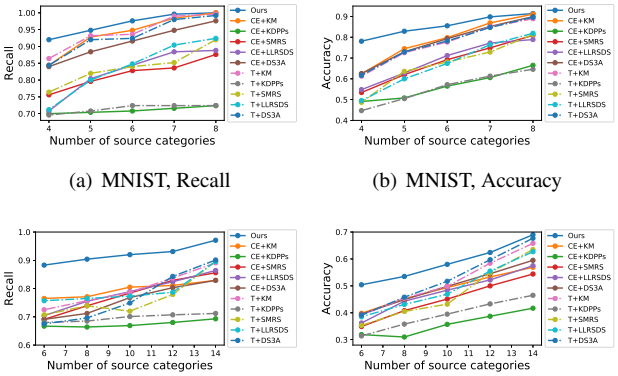
Figure 4 shows the superior performance of our iterative optimization strategy. One obvious limitation of the two-stage strategy is that the feature learning procedure is independent of the target data. In contrast, our iterative strategy can leverage both source data and target data to learn feature representation. Even though target data are unlabeled, the results show that they can still boost the performance significantly. The benefits become more obvious as more unseen categories appear in the target.

4.1.3 Efficiency

Figure 6(a) shows the averaged number of swappings of local search on MNIST dataset. As seen, the proposed local search algorithm can efficiently solve the problem within 20 swappings. For SCENE15 dataset, it can be finished within 25 swappings. Figure 6(b) shows the change of hard triplets and accuracy during the iterative optimization on SCENE15 dataset with 10 categories in the source set. When there are few hard training triplets, further feature learning will cause the oscillation of accuracy. Therefore, the parameter τ can be used to terminate the entire process.

4.1.4 Baseline Comparison

Figure 7 reports the experimental results of our approach and baselines. For the particularly easy MNIST dataset, despite baselines already present strong performances, our method is still able to outperform them. For more complex SCENE15 dataset, our approach can achieve an obvious improvement. On both datasets, the improvement of our proposed method becomes more obvious as more novel categories appear in target data.



(a) MNIST, Recall (b) MNIST, Accuracy

(c) SCENE15, Recall (d) SCENE15, Accuracy

Figure 7. Experimental results of our approach and baselines.

4.2. Key Action Discovery

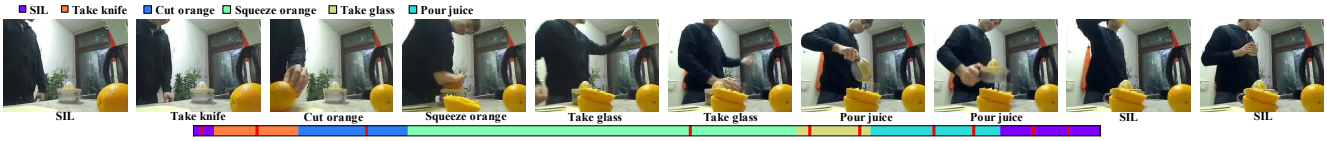
In this section, we evaluate our approach on the task of finding key actions from videos. Two video datasets are used to evaluate the performance, **Breakfast** [18] and **narrated instructional videos** [2]. For Breakfast dataset, there are 1,712 videos with 10 coarse activities (*e.g.*, making *coffee*, *sandwich* and *pancake*) and 48 fine-grained actions. Narrated instructional video dataset includes 5 activities. For each activity, there are 30 videos with 8 to 13 fine-grained key steps.

Experiments are performed on each activity separately and use cross-validation to evaluate the performance: 4-fold for Breakfast based on the provided splits and 5-fold for Narrated dataset. For Breakfast, we use the reduced 64D fisher vector of dense trajectories. For Narrated dataset, we use the provided 3000D bag-of-word feature vectors of the motion and the appearance. In the experiments, we learn a linear transform for better features, *i.e.*, $\mathcal{F}_\theta(x) = Wx$, where $W \in \mathbb{R}^{64\times64}$ for Breakfast and $W \in \mathbb{R}^{256\times3000}$ for Narrated dataset. For both datasets, the source set is constructed by randomly selecting 50% categories. Since Narrated dataset only has 150 videos, in order to mitigate sample bias of randomly sampling, we conduct 5 different runs and report the average results. As the number of key actions in target videos is unknown in advance, we compare the performance as the number of representatives varies within $\{2c, 2.5c, 3c\}$ [1], where c is the number of categories in the source set.

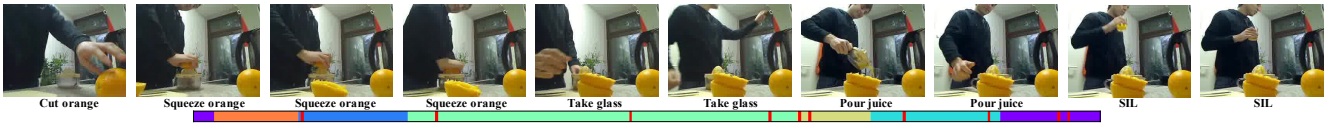
[1]The results of $2c$ and $3c$ can be seen in the supplementary material.

Recall/Accr.	CE+KM	CE+KDPPs	CE+SMRS	CE+LLRSD	CE+DS3A	T+KM	T+KDPPs	T+SMRS	T+LLRSD	T+DS3A	Ours
cereals	0.869 / 0.623	0.816 / 0.562	0.837 / 0.559	0.854 / 0.503	0.798 / 0.571	0.836 / 0.547	0.817 / 0.490	0.834 / 0.472	0.854 / 0.503	0.837 / 0.504	**0.877 / 0.647**
coffee	0.912 / 0.744	0.881 / 0.654	0.905 / 0.676	0.910 / 0.585	0.903 / 0.698	0.899 / 0.642	0.863 / 0.588	0.903 / 0.565	0.910 / 0.585	0.884 / 0.603	**0.935 / 0.767**
friedegg	0.701 / 0.585	0.682 / 0.555	0.702 / 0.542	0.646 / 0.551	0.648 / 0.550	0.691 / 0.559	0.666 / 0.544	0.661 / 0.539	0.646 / 0.551	0.719 / 0.515	**0.750 / 0.627**
juice	0.803 / 0.718	0.709 / 0.652	0.812 / 0.654	0.844 / 0.617	0.669 / 0.651	0.825 / 0.697	0.722 / 0.636	0.804 / 0.563	0.844 / 0.617	0.757 / 0.660	**0.896 / 0.781**
milk	0.863 / 0.598	0.773 / 0.521	0.870 / 0.526	0.852 / 0.472	0.777 / 0.530	0.845 / 0.489	0.808 / 0.442	0.815 / 0.427	0.852 / 0.472	0.821 / 0.449	**0.890 / 0.635**
pancake	0.660 / 0.535	0.659 / 0.501	**0.688** / 0.488	0.630 / 0.500	0.645 / 0.507	0.629 / 0.514	0.643 / 0.484	0.628 / 0.490	0.630 / 0.500	0.687 / 0.476	**0.688 / 0.591**
salat	0.680 / 0.642	0.644 / 0.594	0.707 / 0.567	0.738 / 0.557	0.608 / 0.598	0.751 / 0.618	0.646 / 0.583	0.704 / 0.566	0.738 / 0.557	0.732 / 0.577	**0.796 / 0.685**
sandwich	0.867 / 0.636	0.805 / 0.567	0.876 / 0.573	0.844 / 0.551	0.794 / 0.560	0.866 / 0.568	0.818 / 0.505	0.834 / 0.504	0.844 / 0.551	0.830 / 0.514	**0.890 / 0.693**
scrambledegg	0.763 / 0.574	0.729 / 0.524	0.799 / 0.521	0.737 / 0.514	0.696 / 0.512	0.731 / 0.530	0.733 / 0.500	0.732 / 0.490	0.737 / 0.514	0.727 / 0.484	**0.834 / 0.629**
tea	0.927 / 0.710	0.874 / 0.621	0.914 / 0.647	0.906 / 0.576	0.889 / 0.653	0.905 / 0.600	0.892 / 0.544	0.908 / 0.519	0.906 / 0.576	0.883 / 0.550	**0.928 / 0.749**
Average	0.805 / 0.637	0.757 / 0.575	0.811 / 0.575	0.796 / 0.543	0.743 / 0.583	0.798 / 0.576	0.761 / 0.532	0.782 / 0.514	0.796 / 0.543	0.788 / 0.533	**0.848 / 0.680**

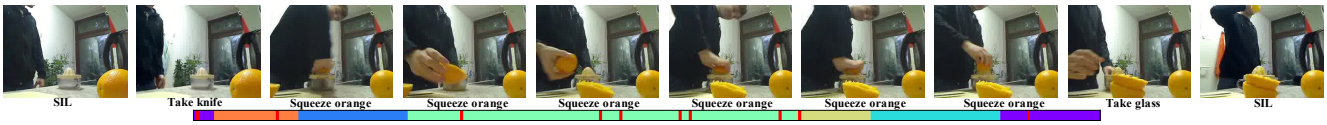
Table 4. Experimental results on Breakfast video dataset with $2.5c$ representatives. c is the number of categories in the source set.



(a) Ours. Recall = 1.000, Accuracy = 0.742



(b) CE+SMRS. Recall = 0.833, Accuracy = 0.390



(c) CE+DS3A. Recall = 0.667, Accuracy = 0.621

Figure 8. Qualitative illustration of representatives in a "juice" video. The location of representatives is highlighted by red lines in the video sequence. The source set only includes four actions, *i.e.*, "SIL", "take knife", "squeeze orange" and "pour juice", while the target set also includes "take glass" and "cut orange".

Quantitive results Table 4 and Table 3 present the experimental results of $2.5c$ representatives on Breakfast and Narrated instructional video dataset respectively. The results show that our approach is able to find the most activities appear in the target videos. Besides, more frames can be represented by the representatives of the same category.

Qualitative results Figure 8 displays picked representatives by our approach, CE+SMRS, and CE+DS3A. Good representatives should achieve both high recall and high accuracy. Figure 8(b) shows that CE+SMRS achieves high recall but low accuracy. The representatives include almost all key actions except for action "*take knife*". But we can find that many representatives are at the boundaries of adjacent actions. Because action boundaries are typically ambiguous, transition frames between two actions actually cannot well represent either action. Figure 8(c) shows that CE+DS3A achieves low recall but high accuracy. It selects many representatives from the long action "*squeeze orange*" and ignores some short actions. Because of the uneven length of actions, ignoring short actions may still achieve a good accuracy but the low recall shows these representatives do not provide good coverage of various categories. Figure 8(a) shows the result of our proposed method, which not only finds all actions but also achieves higher accuracy.

5. Conclusion

In this paper, we design a semi-supervised approach to address the problem of joint representative selection and discriminative feature learning. We leverage the labeled source data to find representatives in the new target data and can discover new categories. Our formulation is based on the facility location problem. We show that the guidance of labeled source data and our proposed discriminative term can effectively improve the performance of both representative selection and feature learning. By iteratively updating the feature representation based on the selected representatives, we show that our strategy can learn better feature for the representative selection than conventional two-stage strategy. Experiments on two image datasets and two video datasets show that our approach not only finds more categories in the target data but also learn better discriminative feature representation for representative selection.

References

[1] R. H. Affandi, A. Kulesza, and E. B. Fox. Markov determinantal point processes. In *UAI*, pages 26–35, 2012. 2

[2] J.-B. Alayrac, P. Bojanowski, N. Agrawal, J. Sivic, I. Laptev, and S. Lacoste-Julien. Unsupervised learning from narrated instruction videos. In *CVPR*, June 2016. 7

[3] V. Arya, N. Garg, R. Khandekar, A. Meyerson, K. Munagala, and V. Pandit. Local search heuristic for k-median and facility location problems. In *STOC*, 2001. 2, 5

[4] J. Bruna and S. Mallat. Invariant scattering convolution networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2013. 6

[5] M. Charikar, S. Guha, E. Tardos, and D. B. Shmoys. A constant-factor approximation algorithm for the k-median problem. In *STOC*, 1999. 2, 5

[6] E. Elhamifar and M. Clara De Paolis Kaluza. Online summarization via submodular and convex optimization. In *CVPR*, July 2017. 2

[7] E. Elhamifar and M. C. De Paolis Kaluza. Subset selection and summarization in sequential data. In *NIPS*, 2017. 1, 2

[8] E. Elhamifar, G. Sapiro, and S. S. Sastry. Dissimilarity-based sparse subset selection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2016. 1, 2, 3, 5

[9] E. Elhamifar, G. Sapiro, and R. Vidal. Finding exemplars from pairwise dissimilarities via simultaneous sparse recovery. In *NIPS*. 2012. 1, 2

[10] E. Elhamifar, G. Sapiro, and R. Vidal. See all by looking at a few: Sparse modeling for finding representative objects. In *CVPR*, 2012. 2, 5

[11] B. J. Frey and D. Dueck. Mixture modeling by affinity propagation. In *NIPS*, 2005. 2

[12] B. J. Frey and D. Dueck. Clustering by passing messages between data points. *Science*, 2007. 1, 2

[13] B. Gong, W.-L. Chao, K. Grauman, and F. Sha. Diverse sequential subset selection for supervised video summarization. In *NIPS*, 2014. 1, 2

[14] M. Gygli and H. G. L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *CVPR*, pages 3090–3098, 2015. 2

[15] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, June 2016. 6

[16] L. Kaufman and P. Rousseeuw. Clustering by means of medoids. In *Statistical Data Analysis Based on the L1 Norm and Related Methods*. 1987. 2, 5

[17] M. R. Korupolu, C. Plaxton, and R. Rajaraman. Analysis of a local search heuristic for facility location problems. *Journal of Algorithms*, 2000. 5

[18] H. Kuehne, A. B. Arslan, and T. Serre. The language of actions: Recovering the syntax and semantics of goal-directed human activities. In *CVPR*, 2014. 7

[19] H. W. Kuhn and B. Yaw. The hungarian method for the assignment problem. *Naval Res. Logist. Quart*, 1955. 5

[20] A. Kulesza and B. Taskar. Structured determinantal point processes. In *NIPS*, 2010. 1

[21] A. Kulesza and B. Taskar. k-dpps: Fixed-size determinantal point processes. In *ICML*, 2011. 2, 5

[22] A. Kulesza and B. Taskar. *Determinantal Point Processes for Machine Learning*. Now Publishers Inc., Hanover, MA, USA, 2012. 2

[23] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, 1998. 5

[24] S. Li and O. Svensson. Approximating k-median via pseudo-approximation. In *STOC*, 2013. 2

[25] J. Meng, H. Wang, J. Yuan, and Y.-P. Tan. From keyframes to key objects: Video summarization by representative object proposal selection. In *CVPR*, June 2016. 2, 5

[26] J. Meng, S. Wang, H. Wang, J. Yuan, and Y. Tan. Video summarization via multiview representative selection. *IEEE Transactions on Image Processing*, 2018. 2

[27] R. Panda, A. Das, Z. Wu, J. Ernst, and A. K. Roy-Chowdhury. Weakly supervised summarization of web videos. In *ICCV*, Oct 2017. 2

[28] A. Prasad, S. Jegelka, and D. Batra. Submodular meets structured: Finding diverse subsets in exponentially-large structured item sets. In *NIPS*, 2014. 1

[29] A. Rodriguez and A. Laio. Clustering by fast search and find of density peaks. *Science*, 2014. 1, 2

[30] C. Schmid. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, pages 2169–2178, 2006. 5

[31] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, June 2015. 4

[32] A. Sharghi, B. Gong, and M. Shah. Query-focused extractive video summarization. In *ECCV*, pages 3–19, 2016. 2

[33] D. B. Shmoys, E. Tardos, and K. Aardal. Approximation algorithms for facility location problems. In *STOC*, 1997. 2

[34] I. Simon, N. Snavely, and S. M. Seitz. Scene summarization for online image collections. In *ICCV*, 2007. 2

[35] S. Tschiatschek, R. K. Iyer, H. Wei, and J. A. Bilmes. Learning mixtures of submodular functions for image collection summarization. In *NIPS*, pages 1413–1421. 2014. 2

[36] L. van der Maaten and G. Hinton. Visualizing high-dimensional data using t-sne. *Journal of Machine Learning Research*, 2008. 6

[37] C. Yang, J. Peng, and J. Fan. Image collection summarization via dictionary learning for sparse representation. In *CVPR*, 2012. 2

[38] Y. Yue and T. Joachims. Predicting diverse subsets using structural svms. In *ICML*, 2008. 1

[39] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman. Video summarization with long short-term memory. In *ECCV*, 2016. 2

[40] B. Zhao and E. P. Xing. Quasi real-time summarization for consumer videos. In *CVPR*, June 2014. 2