Incomplete Multi-View Multi-Label Learning via Disentangled Representation and Label Semantic Embedding

Supplementary Material

6. Complete derivation

6.1. Complete derivation of Eq. (1)

$$
\begin{aligned}
&\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\Big[D_{KL}(q_{\phi_v}(c\mid x^{(v)})\|p(c\mid\{x\}))\Big]\\
=&\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\Big[\int q_{\phi_v}(c\mid x^{(v)})\Big(\log q_{\phi_v}(c\mid x^{(v)})\\
&\qquad-\log p\left(c\mid\{x\}\right)\Big)\mathrm{d}c\Big]\\
=&\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\Big[\int q_{\phi_v}(c\mid x^{(v)})\Big(\log q_{\phi_v}(c\mid x^{(v)})\\
&\qquad-\log p(\{x\}|c)-\log p(c)+\log p(\{x\})\Big)\mathrm{d}c\Big]\\
=&\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\int q_{\phi_v}(c\mid x^{(v)})\log p\left(\{x\}\right)\mathrm{d}c\\
+&\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\Big[\int q_{\phi_v}(c\mid x^{(v)})\log\frac{q_{\phi_v}(c\mid x^{(v)})}{p(c)}\mathrm{d}c\\
&\qquad-\int q_{\phi_v}(c\mid x^{(v)})\log p(\{x\}|c)\mathrm{d}c\Big]\\
=&\log p(\{x\})+\frac{1}{|\mathcal{W}|}\sum_{v\in\mathcal{W}}\Big[D_{KL}(q_{\phi_v}(c\mid x^{(v)})\|p(c))\\
&\qquad-\mathbb{E}_{q_{\phi_v}(c|x^{(v)})}[\log p(\{x\}|c)]\Big]
\end{aligned}
\tag{19}
$$

6.2. Derivation process of Eq. (7)

$$
\begin{aligned}
&I(c^{(v)};s^{(v)};x^{(v)})\\
=&I(c^{(v)};s^{(v)})-I(c^{(v)};s^{(v)}|x^{(v)})\\
=&I(c^{(v)};x^{(v)})-I(c^{(v)};x^{(v)}|s^{(v)})\\
=&I(s^{(v)};x^{(v)})-I(s^{(v)};x^{(v)}|c^{(v)})
\end{aligned}
\tag{20}
$$

The above derivation is based on the definition of Interaction Information [14, 25]. Next, we rearrange the first step of the above derivation to obtain the following formula:

$$
\begin{aligned}
&I(c^{(v)};s^{(v)})\\
=&I(c^{(v)};s^{(v)};x^{(v)})+I(c^{(v)};s^{(v)}|x^{(v)})\\
=&I(c^{(v)};x^{(v)})-I(c^{(v)};x^{(v)}|s^{(v)})+I(c^{(v)};s^{(v)}|x^{(v)})
\end{aligned}
\tag{21}
$$

where $I(c^{(v)};s^{(v)}|x^{(v)})=H(c^{(v)}|x^{(v)})-H(c^{(v)}|x^{(v)},s^{(v)})=H(c^{(v)}|x^{(v)})-H(c^{(v)}|x^{(v)})=0$.

Here, $H(c^{(v)}|x^{(v)},s^{(v)})=H(c^{(v)}|x^{(v)})$ arises from the structural assumption of $q_{\phi_v}(c^{(v)}|x^{(v)})=q_{\phi_v}(c^{(v)}|x^{(v)},s^{(v)})$ [14]. During training, the model disentangles $c^{(v)}$ and $s^{(v)}$ using a disentangling loss (such as the $\mathcal{L}_{disent}$ used in this paper). The goal is to make the consistent feature $c^{(v)}$ and the specific feature $s^{(v)}$ independent of each other. This independence ensures that the extraction of $c^{(v)}$ depends only on $x^{(v)}$, and is independent of $s^{(v)}$. Continuing the derivation from Eq. (21), we obtain:

$$
\begin{aligned}
&I(c^{(v)};s^{(v)})\\
=&I(c^{(v)};x^{(v)})-I(c^{(v)};x^{(v)}|s^{(v)})\\
=&I(x^{(v)};c^{(v)})-I(x^{(v)};c^{(v)}|s^{(v)})\\
=&I(x^{(v)};c^{(v)})+I(x^{(v)};s^{(v)})-I(x^{(v)};c^{(v)},s^{(v)})
\end{aligned}
\tag{22}
$$

7. Additional implementation details

Our DRLS model is developed in Python using PyTorch (version 2.0.1). Across all five datasets, we use a batch size of 128, a learning rate of 0.001, and set d_e to 512. The Adam optimizer is employed during training. In the second phase, the pre-trained model is fine-tuned with a reduced learning rate of 0.0001. All experiments are performed on an NVIDIA RTX 4090 GPU and an Intel i9-13900K CPU. In Fig. 3, we present the structures of the MLP encoder (Fig. 3a), which is used to extract view feature distributions, and the GIN encoder (Fig. 3b), which is used to extract label semantic embedding distributions in our model.

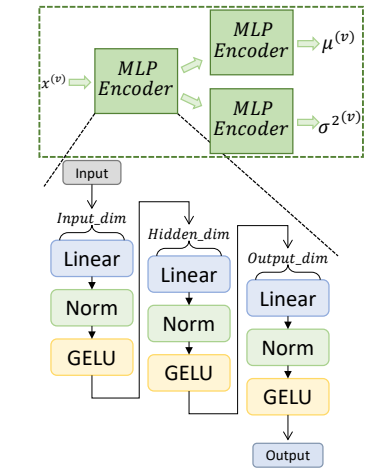
8. Algorithm

The pseudocode for the DRLS model is provided in Algorithm 1.

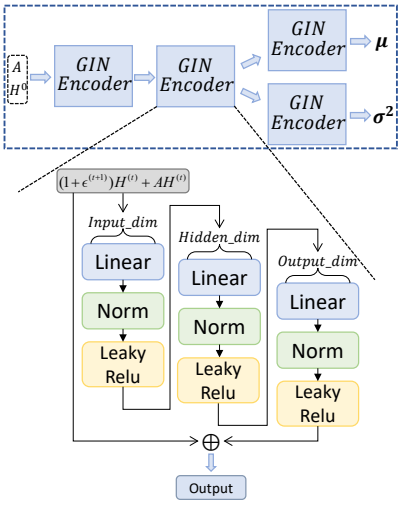
9. Additional experiments

9.1. Missing and training sample rates analysis

In Fig. 4, we show how the proposed method performs under varying missing rates for views and labels. Specifically, Fig. 4a demonstrates the model's performance when the view missing rate changes while the label missing rate remains fixed at 50%. Likewise, Fig. 4b illustrates the impact of different label missing rates on performance, keeping the view missing rate constant at 50%. The results clearly indicate that missing views and labels both degrade the model's effectiveness. As the missing rates increase, the performance gradually declines. However, our model demonstrates adaptability to arbitrary missing scenarios. By comparing Fig. 4a and Fig. 4b, we find that view missing has a



(a) The structure of the MLP encoder



(b) The structure of the GIN encoder

Figure 3. The detailed architectures of the MLP encoder and GIN encoder, where Norm refers to batch normalization, Linear denotes a fully connected layer, and $\oplus$ represents the element-wise addition used in residual connections.

more significant impact on the model's performance, as the model heavily depends on learning features from views.

In Fig. 5, we demonstrate the influence of different training sample rates on Corel5k (Fig. 5a) and Pascal07 (Fig. 5b) datasets. The results show that as the number of training samples increases, the model's performance improves, indicating that more training samples enhance the model's performance.

9.2. Analysis of feature contributions

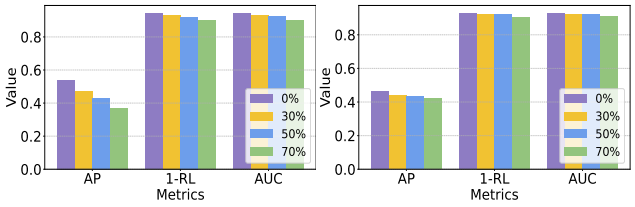
To analyze the contributions of consistent and specific features to model performance, Tab. 5 presents the impact of the fused view-consistent feature $\bar{c}$, the fused view-specific feature $\bar{s}$, and various feature fusion strategies on the model's performance.

As shown in Tab. 5, the fused view-consistent feature $\bar{c}$ contributes more than the fused view-specific feature $\bar{s}$, and the performance of using only multi-view consistent information is inferior to that of fusing consistent and specific information. Furthermore, different feature fusion strategies affect classification performance. Simply summing the features results in insufficient interaction between them, which negatively impacts the model's performance. Applying a

Algorithm 1: The training process of DRLS

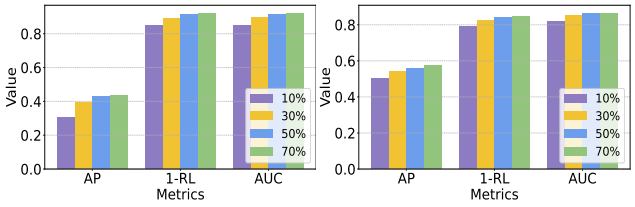
Input: Incomplete multi-view data $\{x^{(v)}\}_{v=1}^{m}$, observable view set $\mathcal{W}$, incomplete multi-label data y, observable label set $\mathcal{U}$, trade-off parameters α, β, and γ, pre-training epochs t_1, and second-phase training epochs t_2.
Output: The trained model parameters.

procedure Pretraining phase
Initialize the pretraining model parameters.
for $t = 1$ **to** t_1 **do**
  Use the encoders $\mu_{\phi_v}(x^{(v)})$ and $\sigma_{\phi_v}(x^{(v)})$ to compute $q_{\phi_v}(c^{(v)}|x^{(v)})$, and reparameterize to derive $c^{(v)}$;
  Use the decoder $p_{\theta_i}(x^{(i)}|c^{(v)})$ to obtain self-view and cross-view reconstructions $\hat{x}^{(i,v)}$;
  Update the model parameters using Eq. (3);
Save $\{q_{\phi_v}(c^{(v)}|x^{(v)})\}_{v=1}^{m}$;
end procedure
procedure The second phase
Initialize the model parameters, compute the adjacency matrix A, and set H^0 as the identity matrix.
for $t = 1$ **to** t_2 **do**
  Reparameterize $q_{\phi_v}(c^{(v)}|x^{(v)})$ and $q_{\varphi_v}(s^{(v)}|x^{(v)})$ to obtain $c^{(v)}$ and $s^{(v)}$;
  Use $p_{\vartheta_v}(x^{(v)}|c^{(v)}, s^{(v)})$ to obtain self-view reconstructions $\hat{x}^{(v)}$;
  Compute $p(c|\{x\})$ using Eq. (5) and reparameterize to obtain the fused feature $\bar{c}$;
  Compute the fused specific feature $\bar{s}$ using Eq. (10);
  Use $\mathrm{GIN}_{\boldsymbol{\mu}}(H^0, A)$ and $\mathrm{GIN}_{\boldsymbol{\sigma}^2}(H^0, A)$ to obtain the semantic embeddings $\{h_i\}_{i=1}^{k}$;
  Reconstruct the adjacency matrix A by taking the inner product of $\{h_i\}_{i=1}^{k}$;
  Use Eq. (11) to compute the fused feature z;
  Obtain $\{r_i\}_{i=1}^{k}$ using $r_i = \omega(f(h_i))$;
  Perform feature selection with Eq. (16) and input the selected features into the classifier to obtain the predictions $\{p_i\}_{i=1}^{k}$;
  Compute the total loss $\mathcal{L}$ as described in Eq. (18) and update the model parameters;
end procedure



(a) Different view missing rates (b) Different label missing rates

Figure 4. The experimental results of DRLS on Corel5k dataset, using 70% of the samples for training, are evaluated with three metrics: AP, 1-RL, and AUC. In particular, (a) presents the performance under varying view missing rates with 50% of the labels missing, while (b) illustrates the performance under varying label missing rates with 50% of the views missing.



(a) Corel5k (b) Pascal07

Figure 5. The experimental results of DRLS on Corel5k (a) and Pascal07 (b) datasets under 50% view missing and 50% label missing conditions with different rates of training samples.

Table 5. The experimental results of DRLS with different features and feature combination methods, evaluated using three typical evaluation metrics and two datasets under conditions of 50% missing views, 50% missing labels, and 70% training samples. In the table, $\omega(\cdot)$ represents the sigmoid activation function, and $\odot$ denotes the Hadamard product.

Features	Corel5k			Pascal07		
	AP	1-RL	AUC	AP	1-RL	AUC
$\bar{c}$	0.419	0.915	0.917	0.561	0.841	0.865
$\bar{s}$	0.386	0.894	0.898	0.539	0.819	0.843
$\bar{s}+\bar{c}$	0.399	0.903	0.906	0.547	0.827	0.850
$\omega(\bar{s})+\bar{c}$	0.425	0.911	0.913	0.551	0.834	0.859
$\bar{s}\odot\bar{c}$	0.421	0.912	0.914	0.565	0.838	0.859
$\omega(\bar{s})\odot\bar{c}$	0.433	0.916	0.918	0.567	0.843	0.864

sigmoid function to the fused view-specific feature $\bar{s}$ is expected to help stabilize the feature distribution, ultimately contributing to improved classification performance.

In summary, multi-view feature fusion strategies are critical for enhancing the model's classification performance. Selecting an appropriate fusion method can further exploit the potential of the features.

9.3. Visualization of the learning process

To verify that our model learns view-consistent features and disentangled view-specific features as intended, we present the cosine similarity heat maps of the features of a randomly selected sample in Figs. 6 to 8. These heat maps depict the cosine similarity between different features at various training epochs, providing an intuitive understanding of how our objective function guides feature learning.

Figs. 6 to 8 are conducted under conditions of 50% missing views and 50% missing labels, where views 1, 2, and 5 of the sample are missing. In Fig. 6, we illustrate the process of learning view-consistent features $\{c^{(v)}\}_{v=1}^{m}$ during the pre-training phase with $\mathcal{L}_{consist}$. As training progresses, the cosine similarity between consistent features from different views steadily increases, indicating that the consistent feature learning aligns with our expectations.

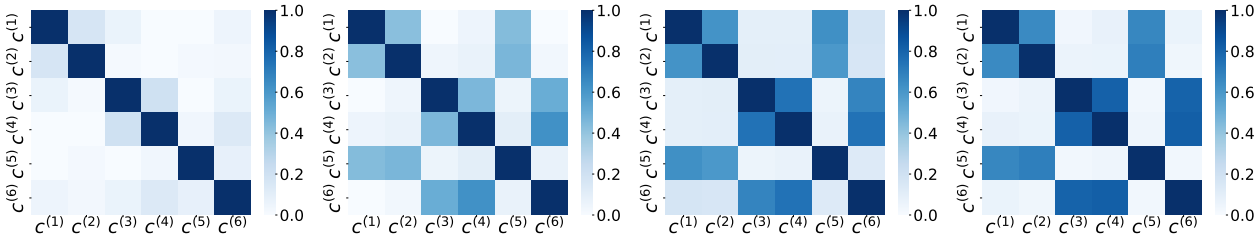
In Fig. 7, we calculate the cosine similarity between consistent features $\{c^{(v)}\}_{v=1}^{m}$ and specific features $\{s^{(v)}\}_{v=1}^{m}$, showing the learning process of view-specific features during the second phase. The similarity between consistent and specific features consistently remains low, confirming that our disentanglement loss $\mathcal{L}_{disent}$ achieves its intended effect. Moreover, at the early stages of training, the similarity between consistent and specific features is already low, which is a result of our pre-training strategy. By learning consistent features in advance, the model captures global consistency information at the early stages of training, laying a solid foundation for subsequent specific feature learning. In Fig. 7, we set β to 1e-2, while in Fig. 8, we increase β to 1e0. Increasing the weight of β raises the contribution of $\mathcal{L}_{disent}$ in the total loss $\mathcal{L}$. Experimental results show that a higher β value indeed leads to better disentanglement. When β = 1e-2, the model achieves the best classification performance on the Corel5k dataset while ensuring effective disentanglement and preserving as much task-relevant information as possible. In this case, the total loss $\mathcal{L}$ is dominated by $\mathcal{L}_{BCE}$, which encourages the model to encode more task-relevant information, leading to a certain degree of similarity among $s^{(v)}$.

9.4. Visualization of classification performance

In Fig. 9 and Fig. 10, we visualize the classification performance of DRLS in comparison to three other advanced models. These experiments are conducted on Corel5k (Fig. 9) and IAPRTC12 (Fig. 10) datasets under conditions of 50% view missing, 50% label missing, and 70% training samples. The figures demonstrate that our method achieves superior performance.

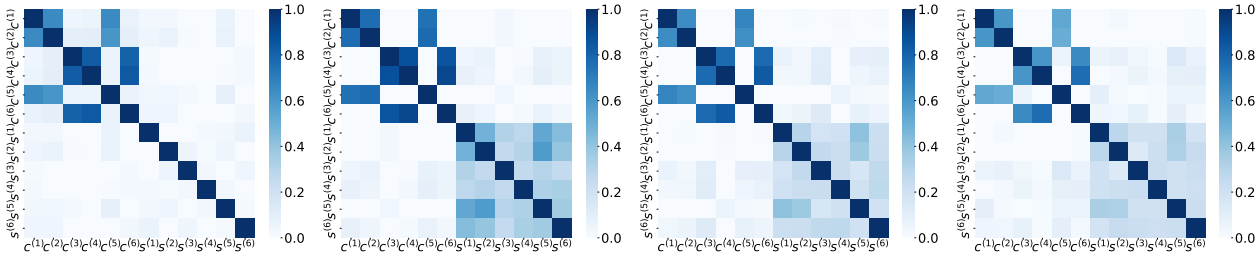
10. Time Complexity Analysis

First, we clarify the notation: n denotes the number of samples, m represents the number of views, k is the number of categories, and $d_{\max}$ refers to the maximum number of neurons in the intermediate network layers. The time complexity of the pre-training phase is $O(nm^2d_{\max}^2 + nm^2)$, where the consistent feature extraction module and the objective function $\mathcal{L}_{consist}$ contribute $O(nm^2d_{\max}^2)$ and $O(nm^2)$,



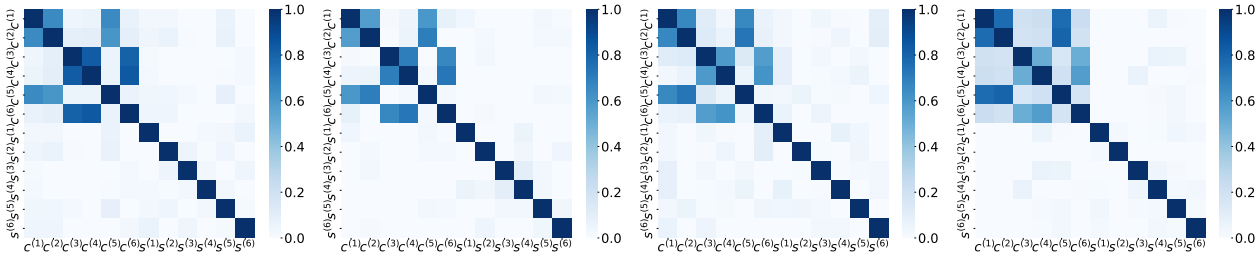
(a) Epoch 0 (b) Epoch 20 (c) Epoch 50 (d) Epoch 90

Figure 6. The cosine similarity heat map of view-consistent features $c^{(v)}$ across different views at various training epochs during the pre-training phase of the DRLS model on Corel5k dataset, under the conditions of 50% view missing, 50% label missing, and 70% training samples.



(a) Epoch 0 (b) Epoch 20 (c) Epoch 50 (d) Epoch 90

Figure 7. The cosine similarity heat map between view-consistent features $c^{(v)}$ and view-specific features $s^{(v)}$ at different training epochs during the second phase of disentangled representation learning of the DRLS model on Corel5k dataset, under the conditions of 50% view missing, 50% label missing, and 70% training samples. Specifically, the trade-off parameter β in the loss function is set to 1e-2.



(a) Epoch 0 (b) Epoch 20 (c) Epoch 50 (d) Epoch 90

Figure 8. Similar to Fig. 7, the heat map during the second training phase is obtained from experiments conducted on Corel5k dataset under conditions of 50% view missing, 50% label missing, and 70% training samples. However, the trade-off parameter β in the loss function is set to 1e0.

respectively. The time complexity of the second phase is $O(nmd^2_{\max} + kd^2_{\max} + nk + nm + k)$, where the disentangled representation learning module and the GIN network contribute $O(nmd^2_{\max})$ and $O(kd^2_{\max})$, respectively. The time complexity of the objective functions $\mathcal{L}_{BCE}$, $\mathcal{L}_{disent}$, and $\mathcal{L}_{le}$ is $O(nk)$, $O(nm)$, and $O(k)$, respectively. The computational cost of DRLS is primarily dominated by the variational autoencoder network.

11. Limitations

Although our method effectively addresses the DIMVMLC problem, it still has some limitations. Currently, we handle missing data by masking it in the loss function. In the future, exploring missing-view recovery techniques could further enhance model performance. For instance, the cross-view reconstruction mechanism we use for learning consistent features may also be leveraged to effectively fill in missing views. Additionally, as analyzed in Sec. 9.3, our method needs to seek a balance between $\mathcal{L}_{disent}$ and $\mathcal{L}_{BCE}$ to achieve optimal results. Future work could explore adaptive strategies to dynamically adjust the disentanglement strength, optimizing this trade-off according to different task requirements.

True labels: ("jet", "plane", "runway")
DICNet: (8, 5, 27)
MTD: (4, 2, 11)
SIP: (5, 4, **3**)
DRLS: (**3**, **1**, 4)

True labels: ("water", "bridge", "arch")
DICNet: (2, 163, 66)
MTD: (7, 15, 18)
SIP: (**1**, 24, 16)
DRLS: (**1**, **7**, **9**)

True labels: ("tree", "snow", "elk")
DICNet: (2, 18, 37)
MTD: (**1**, 82, 86)
SIP: (2, 25, 59)
DRLS: (**1**, **5**, **7**)

True labels: ("sky", "buildings", "light")
DICNet: (181, 6, 18)
MTD: (12, 5, 8)
SIP: (7, 6, **2**)
DRLS: (**6**, **2**, 8)

True labels: ("water", "people", "pool")
DICNet: (23, **1**, 55)
MTD: (**2**, **1**, 7)
SIP: (3, **1**, 7)
DRLS: (**2**, **1**, **4**)

True labels: ("tree", "ice", "field", "frost")
DICNet: (**5**, 8, 13, 12)
MTD: (23, 7, 11, 9)
SIP: (18, 5, 13, 4)
DRLS: (7, **4**, **9**, **2**)

True labels: ("mountain", "sky", "snow")
DICNet: (3, **1**, 14)
MTD: (3, **1**, 15)
SIP: (3, **1**, 10)
DRLS: (**2**, **1**, **5**)

True labels: ("flowers", "tulip", "petals", "stems")
DICNet: (5, 87, 35, 45)
MTD: (**1**, 70, 5, 63)
SIP: (2, 21, 27, 29)
DRLS: (3, **5**, **4**, **9**)

Figure 9. On Corel5k dataset, we present a visual comparison of classification performance across four different methods. The numbers in parentheses for each model represent the likelihood ranking of the sample belonging to each label as predicted by the respective model.

True labels: ("people", "table", "wall")
DICNet: (8, 4, 3)
MTD: (7, 3, **2**)
SIP: (7, 3, 5)
DRLS: (**5**, **2**, 3)

True labels: ("lake", "shore", "woman")
DICNet: (**3**, 4, 7)
MTD: (5, 7, 10)
SIP: (**3**, 4, 10)
DRLS: (5, **3**, **6**)

True labels: ("lake", "shore", "tourist", "tree")
DICNet: (11, 20, 18, 8)
MTD: (14, 25, 15, **6**)
SIP: (6, 13, 32, 7)
DRLS: (**4**, **9**, **10**, 7)

True labels: ("classroom", "desk", "tourist")
DICNet: (7, 8, **6**)
MTD: (20, 10, 8)
SIP: (11, 5, 10)
DRLS: (**6**, **4**, 8)

True labels: ("man", "night", "woman")
DICNet: (**1**, 9, 2)
MTD: (**1**, 12, 2)
SIP: (**1**, 7, 2)
DRLS: (2, **3**, **1**)

True labels: ("landscape", "sky", "tree")
DICNet: (15, 7, 21)
MTD: (8, 2, 21)
SIP: (18, 2, **3**)
DRLS: (**7**, **1**, 5)

True labels: ("beach", "man", "rock", "sea")
DICNet: (19, 10, **2**, 7)
MTD: (14, 28, 3, 12)
SIP: (17, 10, 14, 15)
DRLS: (**6**, **9**, 3, **4**)

True labels: ("field", "grandstand", "player", "spectator")
DICNet: (27, 4, 8, **2**)
MTD: (9, 4, 3, 7)
SIP: (31, 49, 7, 54)
DRLS: (**4**, **2**, **1**, 3)

Figure 10. Similar to the above figure, this shows the results on IAPRTC12 dataset.