Unsupervised Graph Partitioning Framework for Background Suppression in Multi-Query Vehicle Re-Identification

————Appendix————

Appendix A contains all missing proofs in the main manuscript.

Appendix B provides the pseudocode of the proposed GPS framework.

A. Theoretical Proof

Below, we will provide detailed proofs of the theoretical results presented in the methodology Sec. 4.

Notation. First, we recall the notation that we used in the main paper as well as this appendix: The attention map is reinterpreted as an undirected weighted graph $G = (V, A_{\text{sym}})$, where each vertex $v \in V$ corresponds to an image patch, and the edge weights are defined by the symmetrized attention matrix $A_{\text{sym}} = \frac{A+A^\top}{2}$, with A denoting the head-averaged self-attention at layer l. $\mathbf{r} \in [0,1]^N$ denotes the foreground confidence vector, where each element $\mathbf{r}i$ denotes the likelihood that patch i belongs to the foreground. The normalized graph Laplacian is defined as $L = I - D^{-1}A$sym, where $d_i = \sum_j A_{\text{sym}}(i,j)$ is the degree of vertex i. We further define $\mathbf{r}_c = E\mathbf{r}$, where $E = \mathbf{e}_c\mathbf{e}_c^\top$ and $\mathbf{e}c = [1, 0, \ldots, 0]^\top$, and α is a coefficient controlling the strength of the regularization term. The operator $P = D^{-1}A$sym encodes normalized semantic propagation over the attention graph. Finally, the hyperparameter $\beta = \frac{\alpha}{1+\alpha}$ modulates the trade-off between semantic propagation and the anchor constraint.

A.1. Optimality Condition

We aim to compute the gradient of the objective function:

$$J(\mathbf{r}) = \mathbf{r}^\top L\mathbf{r} + \alpha \left\|\mathbf{r}_c - \mathbf{e}_c\right\|_2^2, \tag{1}$$

where $\mathbf{r}_c = E\mathbf{r}$ and $E = \mathbf{e}_c\mathbf{e}_c^\top$ is a rank-one projection matrix.

The first term is a standard quadratic form in $\mathbf{r}$. Since L is symmetric, we have:

$$\nabla_{\mathbf{r}} \left(\mathbf{r}^\top L\mathbf{r}\right) = 2L\mathbf{r}. \tag{2}$$

The second term is a composition of a linear transformation followed by a squared ℓ_2 norm. Let us apply the chain rule:

$$\nabla_{\mathbf{r}} \left\|E\mathbf{r} - \mathbf{e}_c\right\|_2^2 = 2E^\top \left(E\mathbf{r} - \mathbf{e}_c\right). \tag{3}$$

Because E is symmetric ($E^\top = E$) and idempotent ($E^2 = E$), we simplify this to:

$$\nabla_{\mathbf{r}} \left\|\mathbf{r}_c - \mathbf{e}_c\right\|_2^2 = 2E\left(E\mathbf{r} - \mathbf{e}_c\right) = 2\left(E\mathbf{r} - \mathbf{e}_c\right). \tag{4}$$

Putting both components together, we obtain the gradient of J:

$$\nabla_{\mathbf{r}} J(\mathbf{r}) = 2L\mathbf{r} + 2\alpha\left(E\mathbf{r} - \mathbf{e}_c\right) = 2L\mathbf{r} + 2\alpha\left(\mathbf{r}_c - \mathbf{e}_c\right). \tag{5}$$

Setting the gradient to zero yields the optimality condition:

$$\nabla_{\mathbf{r}} J = 2L\mathbf{r}^* + 2\alpha\left(E\mathbf{r} - \mathbf{e}_c\right) = 0. \tag{6}$$

A.2. Fixed-point Equation

We begin with the first-order optimality condition derived in the previous subsection:

$$2L\mathbf{r}^* + 2\alpha\left(E\mathbf{r}^* - \mathbf{e}_c\right) = 0. \tag{7}$$

Divide both sides by 2 and rearrange:

$$L\mathbf{r}^* = -\alpha \left(E\mathbf{r}^* - \mathbf{e}_c\right). \tag{8}$$

Recall that the normalized graph Laplacian is defined as $L = I - P$, where $P = D^{-1}A_{\text{sym}}$ is the normalized propagation operator (row-stochastic and non-negative). Substituting this into the expression, we obtain:

$$\left(I - P\right)\mathbf{r}^* = -\alpha \left(E\mathbf{r}^* - \mathbf{e}_c\right), \tag{9}$$

which rearranges to:

$$\left(I + \alpha E\right)\mathbf{r}^* = P\mathbf{r}^* + \alpha\mathbf{e}_c, \tag{10}$$

To symmetrize the expression and prepare for normalization, we add $\alpha(I - E)\mathbf{r}^*$ to both sides:

$$\left(\alpha + 1\right)\mathbf{r}^* = \left(P + \alpha\left(I - E\right)\right)\mathbf{r}^* + \alpha\mathbf{e}_c. \tag{11}$$

Simplifying the left-hand side:

$$I + \alpha E + \alpha(I - E) = I + \alpha(E + I - E) = I + \alpha I = (1 + \alpha)I. \tag{12}$$

Rewriting the entire expression and obtaining the fixed-point equation:

$$\mathbf{r}^* = \frac{1}{1+\alpha}P\mathbf{r}^* + \frac{\alpha}{1+\alpha}\left[(I - E)\mathbf{r}^* + \mathbf{e}_c\right]. \tag{13}$$

A.3. Convergence Guarantee

We analyze the convergence of the iterative scheme:

$$\mathbf{r}_{t+1} = \frac{1}{1+\alpha}P\mathbf{r}_t + \frac{\alpha}{1+\alpha}\left[(I - E)\mathbf{r}_t + \mathbf{e}_c\right]. \tag{14}$$

with initialization $\mathbf{r}_0 = \mathbf{e}_c$, and $\beta = \frac{\alpha}{1+\alpha} \in (0, 1)$.

This iteration defines a convex combination of two linear operators. Define the operator:

$$T := (1 - \beta)P + \beta(I - E). \tag{15}$$

Since both P and $(I - E)$ are linear, and $\mathbf{e}_c$ is fixed, T is an affine operator.

Step 1: Spectrum of P: We recall that $P = D^{-1}A_{\text{sym}}$ is a row-stochastic matrix:

- A_{sym} is symmetric and non-negative;
- D is the diagonal matrix with $d_i = \sum_j A_{\text{sym}}(i, j)$;
- So each row of P sums to 1, and all entries are non-negative.

By the **Gershgorin Circle Theorem**, all eigenvalues of P lie within the unit disc:

$$|\lambda(P)| \leq \max_i \sum_{j \neq i} P(i, j) < 1. \tag{16}$$

The strict inequality holds because each row of P contains a positive diagonal entry, i.e., $P(i, i) > 0$ for all i, ensuring that the off-diagonal sum is strictly less than 1.

Step 2: Properties of $I - E$: The projection matrix $E = \mathbf{e}_c\mathbf{e}_c^\top$ satisfies $E^2 = E$ and is symmetric, so its eigenvalues are in $0, 1$. Therefore, the matrix $(I - E)$ has eigenvalues in $0, 1$ and is a rank-$(N - 1)$ projection. Its maximum of spectral radius is $\rho(I - E) = 1$.

Step 3: Spectral Radius $\rho(T) < 1$: Since both P and $(I - E)$ are non-negative and linear operators, and $\beta \in (0, 1)$, the matrix T is a convex combination of P and $(I - E)$. Thus, we have:

$$\rho(T) \leq (1 - \beta) \cdot \rho(P) + \beta \cdot \rho(I - E). \tag{17}$$

Recall from Steps 1 and 2: $\rho(P) < 1$, $\rho(I - E) = 1$, So:

$$\rho(T) < (1 - \beta) \cdot 1 + \beta \cdot 1 = 1. \tag{18}$$

This implies that the linear operator T is a strict contraction in spectral norm.

Step 4: Convergence via Banach Fixed-Point Theorem The iteration:

$$\mathbf{r}_{t+1} = T\mathbf{r}_t + \beta\mathbf{e}_c \tag{19}$$

is an affine dynamical system of the form:

$$\mathbf{r}_{t+1} = T\mathbf{r}_t + \mathbf{b}, \quad \text{with } \mathbf{b} = \beta\mathbf{e}_c. \tag{20}$$

Since $\rho(T) < 1$, the operator T is a strict contraction in the Euclidean norm (or any induced norm compatible with the spectral radius). Therefore, by the **Banach fixed-Point theorem**, this iteration converges to a unique fixed point $\mathbf{r}^*$ satisfying:

$$\mathbf{r}^* = T\mathbf{r}^* + \beta\mathbf{e}_c. \tag{21}$$

This fixed point exactly matches the solution derived in the Fixed-point Equation section.

Conclusion: The spectral radius $\rho(T) < 1$ guarantees convergence of the sequence $\{\mathbf{r}_t\}$, and the system stabilizes at a unique solution $\mathbf{r}^*$ regardless of the initial value $\mathbf{r}_0$. In our setup, with $\mathbf{r}_0 = \mathbf{e}_c$, convergence is guaranteed and fast in practice due to contraction strength controlled by $\rho(T)$ and β.

B. Pseudocode of GPS

This section summarizes the proposed Graph Partitioning framework for Background Suppression (GPS) in Algorithm 1. GPS first performs per-view semantic propagation and foreground filtering, then aggregates the filtered tokens across views for a second semantic propagation and filtering stage, and finally outputs a normalized multi-view embedding for retrieval.

Algorithm 1: GPS: Graph Partitioning Framework for Background Suppression

---

Input : Multi-query images $\mathcal{Q} = \{\mathbf{q}^{(k)}\}_{k=1}^{K}$ (with K views per query); ViT backbone ViT with L_{tot} transformer layers; background filtering layer index L; propagation weight β; background ratio ρ; iteration number T

Output : Normalized multi-view embedding $\mathbf{f}^{\text{multi}}$

Let ViT_{ℓ} denote the ℓ-th transformer layer, and $\text{ViT}_{1:L}$, $\text{ViT}_{L+2:L_{\text{tot}}}$ the prefix and tail sub-networks of ViT;

Function `PropagateAndFilter(`A, $[\mathbf{c}, z^1, \ldots, z^N]$, β, ρ, T`)` **:**

  Compute $A_{\text{sym}} = \frac{1}{2}\big(A + A^{\top}\big)$;

  Compute $D = \text{diag}\big(A_{\text{sym}}\mathbf{1}\big)$ and $P = D^{-1}A_{\text{sym}}$;

  Set $\mathbf{e}_c = [1, 0, \ldots, 0]^{\top}$, $E = \mathbf{e}_c\mathbf{e}_c^{\top}$, and initialize $\mathbf{r}_0 = \mathbf{e}_c$;

  for $t = 0$ **to** $T - 1$ **do**

    $\mathbf{r}_{t+1} = (1-\beta)\, P\mathbf{r}_t + \beta\big[(I - E)\mathbf{r}_t + \mathbf{e}_c\big]$;   `// semantic propagation (Eq. (7))`

  Set $\mathbf{r}^* \leftarrow \mathbf{r}_T$ and $\mathbf{r}^*_{\text{patch}} \leftarrow \mathbf{r}^*_{1:N}$;

  Set $K_{\text{keep}} = \lceil (1-\rho)N \rceil$ and $\mathcal{S}_{\text{fg}} \leftarrow \text{TopK}\big(\mathbf{r}^*_{\text{patch}}, K_{\text{keep}}\big)$;

  return $\{\mathbf{c}\} \cup \{z^j \mid j \in \mathcal{S}_{\text{fg}}\}$;

Single-view stage: per-view semantic propagation and filtering;

for $k = 1$ **to** K **do**

  $\mathcal{Z}_L^{(k)}, A^{(k)} \leftarrow \text{ViT}_{1:L}(\mathbf{q}^{(k)})$;

  Decompose $\mathcal{Z}_L^{(k)}$ as $[\,\mathbf{c}^{(k)}, z_L^{1,(k)}, \ldots, z_L^{N,(k)}\,]$;

  $\tilde{\mathcal{Z}}_L^{(k)} \leftarrow$ `PropagateAndFilter`$\big(A^{(k)}, [\,\mathbf{c}^{(k)}, z_L^{1,(k)}, \ldots, z_L^{N,(k)}\,], \beta, \rho, T\big)$;

Multi-view stage: cross-view fusion and second-stage filtering;

$\mathcal{Z}_L^{\text{mv}} \leftarrow \text{concat}\big(\frac{1}{K}\sum_{k=1}^{K}\mathbf{c}^{(k)}, \tilde{\mathcal{Z}}_L^{(1)}, \ldots, \tilde{\mathcal{Z}}_L^{(K)}\big)$;

$\mathcal{Z}_{L+1}^{\text{mv}}, A^{\text{mv}} \leftarrow \text{ViT}_{L+1}(\mathcal{Z}_L^{\text{mv}})$;

Decompose $\mathcal{Z}_{L+1}^{\text{mv}}$ as $[\,\bar{\mathbf{c}}, z_{L+1}^{1,\text{mv}}, \ldots, z_{L+1}^{N_{\text{mv}},\text{mv}}\,]$;

$\mathcal{Z}_{\text{final}} \leftarrow$ `PropagateAndFilter`$\big(A^{\text{mv}}, [\,\bar{\mathbf{c}}, z_{L+1}^{1,\text{mv}}, \ldots, z_{L+1}^{N_{\text{mv}},\text{mv}}\,], \beta, \rho, T\big)$;

Final embedding;

$\mathbf{f}^{\text{multi}} \leftarrow \text{ViT}_{L+2:L_{\text{tot}}}(\mathcal{Z}_{\text{final}})$;   `// embedding from the final [CLS] token`

$\mathbf{f}^{\text{multi}} \leftarrow \mathbf{f}^{\text{multi}} / \|\mathbf{f}^{\text{multi}}\|_2$;

return $\mathbf{f}^{\text{multi}}$;
