Reference Material

This reference material contains the derivation details for obtaining the distance matrix by learning constrained transformation of feature observations as presented in Section III-B of the paper. It also includes the details of the optimization solving for the learning of correlation filters combined with fine-tuning of the distance matrix in Section III-C.

A. Learning Constrained Transformation

Here the derivation of the learning process for constrained transformations of feature observations is supplemented. A parameterized feature-map distance $d_c^2(f_i, f_j)$ was introduced in Eq. (5). In this section, we aim to learn a distance matrix $M^{N\times N}$ under correlation constraints. Intuitively, the goal is for M to capture a transformed distance that remains close to the intrinsic distances given by the feature map $f_{N\times 1}$, while also respecting certain upper/lower bounds imposed by feature-map relationships. Mathematically, we formulate this as follows.

1) Motivation

To make the learned distance transformation align well with the underlying feature map itself, we focus on minimizing the discrepancy between two covariance-like matrices, namely those represented by $p(f;\mu,C)$ and $p(f;\mu,M)$. Here, $C\in\mathbb{R}^{N\times N}$ is a covariance-like matrix associated with the "true" feature distance, whereas $M\in\mathbb{R}^{N\times N}$ is to be learned. We further incorporate correlation constraints arising from the distance bounds among feature pairs.

2) Distribution Formulation

Since a periodically repeated Gaussian function was used in the interpolation of the detection scores $S_{M,\tau}\{x\}$, we represent the feature-map distribution via a multivariate Gaussian. Let $\mu=\mu_{N\times 1}=(\mu_1,\mu_2,\dots,\mu_N)$ be the mean vector of the feature maps $f_{N\times 1}$. Then we specify

$$p(f;\mu,M)=\frac{1}{\sqrt{(2\pi)^N\,|M|}}exp(-\frac{1}{2}d_m^2(f,\mu)), \tag{A.1}$$

which parallels the parametric form

$$p(f;\mu,C)=\frac{1}{\sqrt{(2\pi)^N\,|C|}}\exp\left(-\frac{1}{2}(f-\mu)^\top C^{-1}(f-\mu)\right).$$

3) KL-Divergence Objective

We treat $p(f;\mu,C)$ as the true distribution and $p(f;\mu,M)$ as the fitted distribution we wish to learn. The discrepancy is thus measured by the relative entropy (KL divergence):

$$R(p(f;\mu,C)\parallel p(f;\mu,M))=\sum_{k=1}^{N}p(f_k;\mu_k,C)\cdot\log\frac{p(f_k;\mu_k,C)}{p(f_k;\mu_k,M)}. \tag{A.2}$$

Where f_k iterates over feature samples. From the properties of Gaussian distributions, one can show that the KL divergence admits a concise closed-form:

$$R(p(f;\mu,C)\parallel p(f;\mu,M))=\frac{1}{2}(\log\frac{|M|}{|C|}+tr(M^{-1}C)-N). \tag{A.3}$$

This result is closely related to Stein's loss in comparing covariance matrices. The key algebraic identity used here is $\lambda^\top K\,\lambda=\mathrm{tr}\big(K\,\lambda\lambda^\top\big)$, which allows rewriting vector norms in a trace form. This property ensures scale invariance and conveniently unifies certain distance constraints with matrix inequalities.

4) Distance Constraints

We also impose upper/lower bound constraints on the distances among feature pairs: for certain pairs $(i,j)\subseteq S^{(1)}$ or $(i,j)\subseteq S^{(2)}$, the transformed distances $d_c^2(f_i,f_j)$ must be below thresholds ς_1,ς_2, whereas for cross-set pairs $\big(i\in S^{(1)}, j\in S^{(2)}\big)$, it must exceed some margin ϱ. Here, $S^{(1)}$ and $S^{(2)}$ are two sets of feature indices that we wish to keep close (intra-set) versus pushed apart (inter-set). Formally, these constraints yield:

$$\begin{aligned}\min_{M\succeq 0}\quad & \sum_{k=1}^{N}p(f_k;\mu_k,C)\log\frac{p(f_k;\mu_k,C)}{p(f_k;\mu_k,M)}\\ s.t.\quad & d_m^2(f_i,f_j)\leqslant\varsigma_1,\ (i,j)\subseteq S^{(1)};\\ & d_m^2(f_i,f_j)\leqslant\varsigma_2,\ (i,j)\subseteq S^{(2)};\\ & d_m^2(f_i,f_j)\geqslant\varrho,\ \ i\in S^{(1)},j\in S^{(2)}.\end{aligned} \tag{A.4}$$

By combining the KL objective with these geometric constraints, we seek an M that aligns with C but also enforces the desired cluster separation or proximity among different feature subsets.

5) Trace-Based Formulation

Substituting Eq. (4), Eq. (5), Eq. (A.1) and the Gaussian parameterization of C into Eq. (A.2), and exploiting the identity $\lambda^\top K\,\lambda=\mathrm{tr}\big(K\,\lambda\lambda^\top\big)$, we transform all constraints into linear forms in M. Specifically, we rewrite $d_c^2(f_i,f_j)$ as $\mathrm{tr}\big(M\,(f_i-f_j)\,(f_i-f_j)^\top\big)$. Hence, Eq. (A.4) is turned into:

$$\begin{aligned}\min_{M\succeq 0}\quad & \frac{1}{2}(\log\frac{|M|}{|C|}+tr(M^{-1}C)-N)\\ s.t.\quad & tr(M(f_i-f_j)(f_i-f_j)^\top)\leqslant\varsigma_1,\ (i,j)\subseteq S^{(1)};\\ & tr(M(f_i-f_j)(f_i-f_j)^\top)\leqslant\varsigma_2,\ (i,j)\subseteq S^{(2)};\\ & tr(M(f_i-f_j)(f_i-f_j)^\top)\geqslant\varrho,\ \ i\in S^{(1)},j\in S^{(2)}.\end{aligned} \tag{A.5}$$

In rare cases, no solution to Eq. (A.5) exists (e.g. because the sets $S^{(1)}$ and $S^{(2)}$ impose incompatible constraints). Therefore, we introduce an additional slack-matrix variable $\mathrm{diag}(\xi)$ to relax the constraints if necessary.

6) Slack Variable and Final Formulation

The modified problem becomes:

$$\begin{aligned}\min_{M\succeq 0,\xi}\quad & \frac{1}{2}(\log\frac{|M|}{|C|}+tr(M^{-1}C)-N)+\kappa(\log\frac{|diag(\xi)|}{|diag(\xi_0)|}\\ & +tr(diag(\xi)diag(\xi_0))-N)\\ s.t.\quad & tr(M(f_i-f_j)(f_i-f_j)^\top)\leqslant\varsigma_1,\ (i,j)\subseteq S^{(1)};\\ & tr(M(f_i-f_j)(f_i-f_j)^\top)\leqslant\varsigma_2,\ (i,j)\subseteq S^{(2)};\\ & tr(M(f_i-f_j)(f_i-f_j)^\top)\geqslant\varrho,\ \ i\in S^{(1)},j\in S^{(2)},\end{aligned} \tag{A.6}$$

where $\mathrm{diag}(\xi)$ is initialized by $\mathrm{diag}(\xi_0)$. Depending on which constraint each pair (i,j) belongs to, ξ_0 is set to ς_1, ς_2, or ϱ.

The parameter $\kappa > 0$ balances how strictly we try to satisfy the distance constraints versus minimizing the KL divergence. A larger κ devotes more effort to enforcing feasibility, while a smaller κ focuses on aligning M and C.

7) Lagrangian Method and Iterative Update

We solve Eq. (A.6) via an extended Lagrangian multiplier approach, leading to the following iterative update for the matrix M:

$$M_{t+1} = (1 + \psi M_t (f_i - f_j)(f_i - f_j)^\top) M_t, \tag{A.7}$$

where ψ is the current Lagrange multiplier that depends on whether a given constraint is of the "push together" (ς_1, ς_2) type or the "push apart" (ϱ) type. Specifically, ψ evaluates to $\frac{-\alpha}{1+\alpha\, d^2_{M_t}}$ for $i \in S^{(1)}, j \in S^{(2)}$, and $\frac{\alpha}{1-\alpha\, d^2_{M_t}}$ otherwise, where $\alpha = \frac{1}{2}\big(\frac{1}{d^2_{M_t}} - \frac{\kappa}{\xi}\big)$ and $d^2_{M_t} = \mathrm{tr}\big(M_t\,(f_i - f_j)\,(f_i - f_j)^\top\big)$. Repeatedly applying Eq. (A.7) for T_1 iterations yields a final solution for M.

The above formulation formally establishes the metric-like behavior for feature similarities, where M is learned to respect upper/lower distance constraints. Should the problem be infeasible, the slack variables adjust the constraints to ensure that a solution emerges. This paves the way for a learned transformation matrix M that captures essential feature distances in a constrained yet robust manner.

B. Joint Training with Correlation Filters

Here the procedure of solving optimization problems for online learning regarding correlation filters is supplemented, which unites the fine-tuning of the distance matrix.

1) Motivation and Objective

To reduce redundancy in the feature space, we perform eigenvalue decomposition (or equivalently, PCA) on the matrix $M^{N\times N}$ obtained in Section III-B), then truncate several principal eigenvectors to derive a lower-dimensional projection matrix $\tilde{M}^{N\times V}$. Let $\tau = (\tau^1, \tau^2, \ldots, \tau^V)$ be a set of discriminative correlation filters (DCF). We jointly fine-tune $\tilde{M}$ and τ in a single image frame by minimizing the following objective:

$$L(\tilde{M}, \tau) = \left\| G - \sum_{v=1}^{V} F_v\{x^v\}\, \tilde{M} \circledast \tau^v \right\|_2^2 + \omega \sum_{v=1}^{V} \|\tau^v\|_2^2 + \gamma \|\tilde{M}\|_2^2, \tag{B.1}$$

where G is the desired detection score (i.e., labeled response), $\tilde{M}^{N\times V}$ is the projection matrix, τ^v denotes the v-th filter channel, and ω and γ are the regularization parameters. For simplicity, Eq. (B.1) is formulated for one frame, but in practice it may be extended over multiple frames.

2) Frequency-Domain Reformulation

Following conventional correlation filter approaches, we transform Eq. (B.1) into the frequency domain via Parseval's formula. Let $\hat{f}$ be the discrete Fourier transform (DFT) of f. Then Eq. (B.1) becomes:

$$\hat{L}(\tilde{M}, \tau) = \left\| \hat{G} - \sum_{v=1}^{V} \hat{F}_v\{x^v\}\, \tilde{M} \odot \hat{\tau}^v \right\|_2^2 + \omega \sum_{v=1}^{V} \|\hat{\tau}^v\|_2^2 + \gamma \|\tilde{M}\|_2^2, \tag{B.2}$$

where $\odot$ denotes element-wise (Hadamard) multiplication, $\hat{F}_v\{x^v\}$ is the frequency-domain representation of the v-th feature channel, and $\hat{\tau}^v$ is the DFT of the filter τ^v. The objective $\hat{L}$ is an unconstrained nonlinear least-squares problem, showing a near bilinear form in terms of $\tilde{M}$ and τ.

3) First-Order Approximation

To solve for $\tilde{M}$ and τ iteratively, we adopt a Gauss–Newton or first-order Taylor expansion approach. Denote the current estimates at iteration t by $(\tilde{M}_t, \hat{\tau}_t)$. Let $\Delta\tilde{M}_t$ and $\Delta\hat{\tau}_t$ be the increments. Consider the frequency-domain product

$$\hat{F}\big(\tilde{M}_t + \Delta\tilde{M}_t\big)\,\big(\hat{\tau}_t + \Delta\hat{\tau}_t\big).$$

A first-order Taylor expansion around $(\tilde{M}_t, \hat{\tau}_t)$ gives:

$$\hat{F}\,\big(\tilde{M}_t + \Delta\tilde{M}_t\big)\,\big(\hat{\tau}_t + \Delta\hat{\tau}_t\big) \approx \underbrace{\hat{F}\,\tilde{M}_t\,\hat{\tau}_t}_{\text{current solution}} + \underbrace{\hat{F}\,\tilde{M}_t\,\Delta\hat{\tau}_t}_{\text{linear in }\Delta\hat{\tau}_t} + \underbrace{\big(\hat{\tau}_t^\top \otimes \hat{F}\big)\,\Delta\tilde{M}_t'}_{\text{linear in }\Delta\tilde{M}_t}, \tag{B.3}$$

where $\Delta\tilde{M}_t'$ is the vectorization of $\Delta\tilde{M}_t$ (stacking all channels and spatial positions), and $\otimes$ is the Kronecker product for coupling $\Delta\tilde{M}_t$ and $\hat{\tau}_t$. To simplify notation, let

$$\hat{\tau}_{t,\Delta} \;\equiv\; \hat{\tau}_t + \Delta\hat{\tau}_t.$$

We can incorporate the constant term $\hat{F}\,\tilde{M}_t\,\hat{\tau}_t$ into the objective residual. Thus, we focus on the linearized increments $\Delta\tilde{M}_t$ and $\Delta\hat{\tau}_t$.

4) Quadratic Subproblem

Substituting Eq. (B.3) into Eq. (B.2) while ignoring second-order small quantities, we obtain a quadratic subproblem in terms of $\Delta\tilde{M}_t$ and $\Delta\hat{\tau}_t$. For notational convenience, define

$$\text{Æ}_{\tilde{M}} = \big[\text{Æ}^1_{\tilde{M}}, \text{Æ}^2_{\tilde{M}}, \ldots, \text{Æ}^V_{\tilde{M}}\big], \quad \text{Œ}_\tau = \begin{pmatrix} (\hat{\tau}_t^\top \otimes \hat{F})[-Q] \\ \vdots \\ (\hat{\tau}_t^\top \otimes \hat{F})[Q] \end{pmatrix},$$

where $\text{Æ}^v_{\tilde{M}}$ is obtained by placing $\hat{F}[k]\tilde{M}_t^v$ in a diagonal matrix and padding zeros (to align different channels of size Q vs. Q_v), with $Q = \max(Q_v)$ and $Q_v = \lfloor D_v/2 \rfloor$. The linearized objective then becomes:

$$\tilde{L}\big(\Delta\tilde{M}_t', \hat{\tau}_{t,\Delta}\big) = \left\| \text{Æ}_{\tilde{M}}\,\hat{\tau}_{t,\Delta} + \text{Œ}_\tau\,\Delta\tilde{M}_t' - \hat{G} \right\|_2^2 + \omega\,\|\hat{\tau}_{t,\Delta}\|_2^2 + \gamma\,\|\tilde{M}_t + \Delta\tilde{M}_t'\|_2^2, \tag{B.4}$$

where $\tilde{M}_t^v$ indicates the v-th channel of $\tilde{M}_t$ (vectorized in the frequency domain). In practice, one typically stacks all channels of $\tilde{M}_t$ into a single long vector for an iterative update.

5) Normal Equation

Setting the gradients of Eq. (B.4) with respect to $\hat{\tau}_{t,\Delta}$ and $\Delta\tilde{M}_t'$ to zero yields the following block-structured linear system:

$$\begin{bmatrix} \text{Æ}^H_{\tilde{M}}\,\text{Æ}_{\tilde{M}} + \omega I & \text{Æ}^H_{\tilde{M}}\,\text{Œ}_\tau \\ \text{Œ}^H_\tau\,\text{Æ}_{\tilde{M}} & \text{Œ}^H_\tau\,\text{Œ}_\tau + \gamma I \end{bmatrix} \begin{bmatrix} \hat{\tau}_{t,\Delta} \\ \Delta\tilde{M}_t' \end{bmatrix} = \begin{bmatrix} \text{Æ}^H_{\tilde{M}}\,\hat{G} \\ \text{Œ}^H_\tau\,\hat{G} \;-\; \gamma\,\tilde{M}_t^v \end{bmatrix}, \tag{B.5}$$

where $\text{Æ}^H_{\tilde{M}}$ and Œ^H_τ are the conjugate transposes, and $\tilde{M}_t^v$ denotes the vectorized form of $\tilde{M}_t$ for channel v (brought in by the regularization term $\|\tilde{M}\|_2^2$ upon linearization). Equation Eq. (B.5) can be solved by conjugate gradient or other iterative methods to obtain the increments $\Delta\tilde{M}_t, \Delta\hat{\tau}_t$, which then update $\tilde{M}$ and τ for the next iteration.

The above procedure jointly trains the correlation filters τ and refines the projection $\tilde{M}$. The result is a more discriminative set of filter templates and a constrained projection matrix, both of which aim to enhance the tracking performance. If computational cost becomes excessive, one may reduce the number of principal components in $\tilde{M}$ or limit the iteration count to balance accuracy and efficiency.