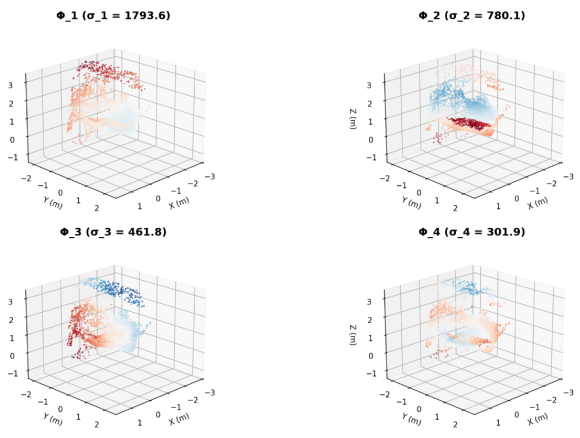
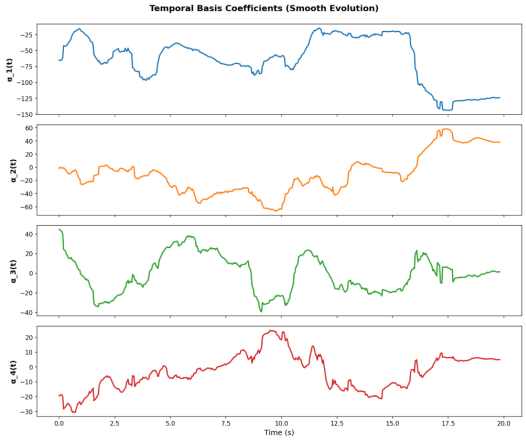
Temporally-Smooth Global Bundle Adjustment for Real-Time Dense Visual SLAM

Supplementary Material

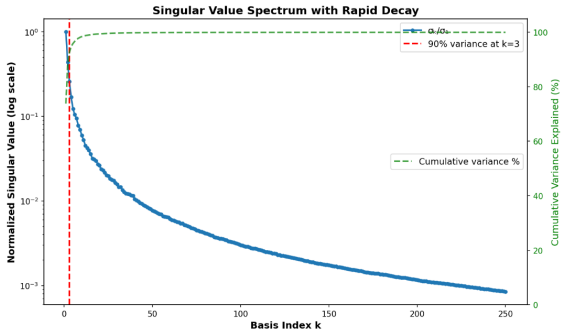
A. Temporal Basis Visualization



(Panel a)



(Panel b)



(Panel c)

Figure 1. Temporal basis visualization: (a) First four basis functions $\mathbf{\Phi}_1$–$\mathbf{\Phi}_4$ as 3D point clouds, (b) Coefficient evolution $\alpha_k(t)$ showing smooth temporal coherence, and (c) Singular value decay σ_k/σ_1 with "elbow" at $K \approx 250$.

B. Real-time Global Bundle Adjustment Algorithm

Algorithm 1 Real-time Global Bundle Adjustment

Require: Video frames $I_1, I_2, \ldots, I_N$
Ensure: Camera poses $\mathbf{T}$, dense reconstruction $\mathbf{X}(t)$

▷ Initialization
$\mathbf{T}_0, \mathbf{T}_1, \mathbf{X}_0 \leftarrow \textsc{TwoViewSfM}(I_0, I_1)$
$\Phi \leftarrow \{\mathbf{X}_0\}$ ▷ Initial basis (mean geometry)
$\mathcal{K} \leftarrow \{0, 1\}$ ▷ Initial keyframe set
▷ Main loop
for each frame I_i, $i = 2, \ldots, N$ **do**
  ▷ Thread 2: Feature extraction
  $\mathbf{f}_i \leftarrow \textsc{ExtractDenseFeatures}(I_i)$
  matches $\leftarrow \textsc{MatchFeatures}(\mathbf{f}_i, \mathbf{f}_{i-1})$
  ▷ Thread 3: Quick tracking (warm start)
  $\mathbf{T}_i^{(0)}, \boldsymbol{\alpha}_i^{(0)} \leftarrow \textsc{Predict}(\mathbf{T}_{i-1}, \boldsymbol{\alpha}_{i-1})$ ▷ Eq. (14)–(15)
  $\boldsymbol{\alpha}_i \leftarrow \textsc{OptimizeCoefficients}(\text{matches}, \Phi, \mathbf{T}_i^{(0)})$ ▷ Eq. (10)
  ▷ Thread 4: Global pose optimization (continuous)
  for iter $= 1$ to max_iterations **do**
    $\mathbf{J}_\mathbf{T}, \mathbf{J}_{\boldsymbol{\alpha}} \leftarrow \textsc{ComputeJacobians}(\mathbf{T}, \boldsymbol{\alpha}, \Phi)$ ▷ GPU parallel
    $\mathbf{H}, \mathbf{b} \leftarrow \textsc{FormNormalEquations}(\mathbf{J}_\mathbf{T}, \mathbf{J}_{\boldsymbol{\alpha}})$ ▷ GPU parallel
    $\mathbf{S}, \mathbf{b}_{\text{reduced}} \leftarrow \textsc{SchurComplement}(\mathbf{H}, \mathbf{b})$ ▷ Eq. (11)
    $\Delta\mathbf{T} \leftarrow \textsc{SolveSparse}(\mathbf{S}, \mathbf{b}_{\text{reduced}})$ ▷ CPU Cholesky
    $\mathbf{T} \leftarrow \textsc{UpdatePoses}(\mathbf{T}, \Delta\mathbf{T})$
    if converged **then**
      break
    end if
  end for
  ▷ Keyframe decision and basis update
  if $\textsc{IsKeyframe}(\mathbf{T}_i, \text{matches})$ **then** ▷ Eq. (12)
    $\mathcal{K} \leftarrow \mathcal{K} \cup \{i\}$
    ▷ Thread 5: Incremental basis update
    $\mathbf{X}_i \leftarrow \mathbf{X}_0 + \sum_k \alpha_{ik} \cdot \Phi_k$
    $\Phi \leftarrow \textsc{IncrementalSVD}(\Phi, \mathbf{X}_i - \mathbf{X}_0)$ ▷ Eq. (5)
    $\textsc{PruneBasis}(\Phi, \text{threshold} = 10^{-4})$
    if $|\mathcal{K}| > 500$ **then**
      $\textsc{MarginalizeOldKeyframes}(\mathcal{K}, \Phi)$ ▷ Eq. (13)
    end if
  end if
  ▷ Thread 1: Dense reconstruction (async)
  $\mathbf{X}_{\text{dense}} \leftarrow \mathbf{X}_0 + \sum_k \alpha_{i,k} \cdot \Phi_k$
  $\textsc{Visualize}(\mathbf{X}_{\text{dense}})$
end for

C. EuRoC Trajectory Accuracy

Table 1. ATE (cm) on EuRoC. Best in **bold**.

Sequence	Ours	ORB SLAM3	DROID SLAM	NICE SLAM
V1_01_easy	**2.8**	3.9	4.4	7.2
V1_02_medium	**3.1**	4.2	4.9	8.1
V1_03_difficult	4.2	5.8	**3.8**	9.6
V2_01_easy	**2.9**	4.1	5.6	7.8
V2_02_medium	3.4	4.8	**3.1**	8.9
V2_03_difficult	4.1	6.2	**3.9**	10.2
MH_01_easy	**2.6**	3.4	4.2	6.8
MH_02_easy	**2.7**	3.6	3.3	7.1
MH_03_medium	**3.2**	4.5	4.9	8.4
MH_04_difficult	**3.8**	5.3	6.4	9.8
MH_05_difficult	3.4	4.9	**3.1**	9.2
Average	**3.3**	4.6	4.8	8.5