KALMAN FILTER-BASED TEMPORAL PROMPTING FOR VIDEO DEPTH COMPLETION

Anonymous authors
Paper under double-blind review

ABSTRACT

3D depth sensors provide accurate measurements, yet recovering dense, temporally consistent depth from these sparse observations is crucial. Existing single-frame depth completion methods suffer from temporal inconsistencies due to frame-wise processing, whereas recent video depth models incur substantial computational overhead, limiting their practical deployment. To address this limitation, we propose a lightweight framework for adapting foundation models using visual prompts, avoiding the need for full retraining or computationally intensive temporal designs. To enhance temporal consistency across multiple frames, we propose a Kalman Filter-based Temporal Prompting (KFTP), which models temporal relationships by fusing the previous and current visual prompts. Furthermore, we design uncertainty-aware prompting with adaptive process noise that explicitly models temporal uncertainty to prevent error accumulation. This enables robust adaptation under dynamic motion, occlusions, and highly sparse supervision.

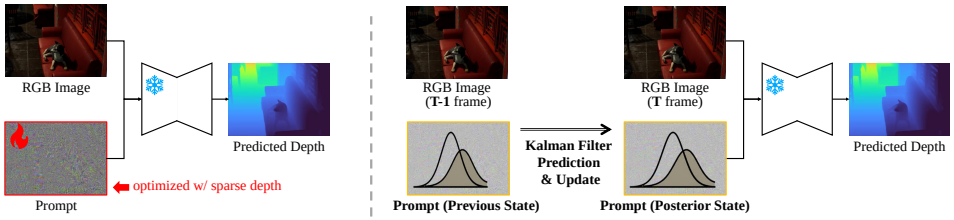
1 INTRODUCTION

Video depth estimation is fundamental to autonomous driving (Park et al., 2021; Wang et al., 2019), robotics (Dong et al., 2022), and augmented reality (Holynski & Kopf, 2018). In this context, video depth completion addresses the problem of reconstructing dense depth maps from sparse observations (e.g., LiDAR or structured-light measurements) over continuous video streams. Although the 3D sensors provide metrically accurate depth, their measurements are inherently limited in spatial coverage, yielding observations at only a small subset of pixels in each frame. Consequently, video depth completion propagates and regularizes these sparse cues to produce depth maps that are geometrically coherent and temporally stable, enabling reliable 3D perception.

Recent single-frame depth completion methods have increasingly focused on generalizable and zero-shot settings, where pretrained depth models are adapted to sparse sensor measurements without dataset-specific retraining (Hyoseok et al., 2025; Jeong et al., 2025; Viola et al., 2025). These approaches demonstrate strong spatial accuracy and computational efficiency across diverse scenes by leveraging robust monocular depth priors. However, they process each frame independently and therefore cannot exploit temporal information from continuous video streams. As a result, even if each frame is reasonably accurate, small frame-wise errors can fluctuate over time in dynamic scenes with ego-motion, parallax, and occlusions, producing flicker and frame-to-frame jitter. In contrast, generalizable video depth completion remains largely underexplored. Existing video-based methods attempt to incorporate temporal cues through adjacent-frame feature aggregation or ray-based cost-volume fusion (Zhu et al., 2025; Kim et al., 2024), but they are typically designed for specific sensor configurations or environments. This limits their ability to generalize to heterogeneous sensors and unseen domains, leaving zero-shot video depth completion an open and practical challenge.

To achieve generalizable depth completion across a variety of sensors and environments, we leverage pre-trained depth foundation models, which provide strong geometric priors learned from large-scale data, and adapt them efficiently at test time rather than retraining per environment. To do this, we adopt visual prompt tuning as a lightweight adaptation mechanism, in which the foundation model remains frozen while the input visual prompt is optimized during inference to align the model with the target sensor and scene. The most relevant work (Jeong et al., 2025) performs this optimization independently per frame, where temporal information is not exploited, and the predictions often flicker across frames.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.



(a) Single image Visual Prompt (b) Kalman Filter-based Temporal Visual Prompt

Figure 1: (a) Single-image visual prompting optimizes the prompt independently for each frame, which can cause temporally inconsistent predictions. (b) KFTP models the prompt as a time-evolving state and fuses the propagated prior with a frame-specific observation through uncertainty-aware Kalman prediction and update.

To address these limitations, we propose Kalman Filter-based Temporal Prompting (KFTP), which models the visual prompt as a time-evolving state (Figure 1) and estimates it with the prediction–update recursion of the Kalman filter (Kalman, 1960) combining motion-guided prompt propagation with uncertainty-aware prompt fusion. The framework begins with a prediction step that propagates the prior prompt using temporal relationships. To capture uncertainty arising from scene dynamics, we introduce an adaptive process-noise term that accounts for feature uncertainty, geometric error, and motion magnitude, preventing overconfident propagation. In the update step, we refine the propagated prompt via test-time optimization against the current sparse measurements and treat the optimized prompt as a noisy observation. The Kalman update then fuses this observation with the propagated prior to obtain a posterior prompt, which is fed into the frozen depth foundation model to produce dense depth predictions for each frame. Experiments on indoor and outdoor benchmarks show that KFTP consistently improves temporal stability while maintaining competitive spatial accuracy, supporting its practicality for real-world streaming deployment.

2 Related Work

2.1 Video Depth Estimation

Video depth estimation aims to generate temporally consistent depth maps from monocular sequences. Early works extend single-image models with recurrent units, optical-flow warping, or feature aggregation to incorporate temporal information (Luo et al., 2020; Patil et al., 2020; Zhang et al., 2019; 2021), and more recent approaches design multi-frame fusion networks that combine neighboring-frame features to suppress flickering (Hu et al., 2025; Shao et al., 2025; Chen et al., 2025; Chou et al., 2025; Zhang et al., 2025; Piccinelli et al., 2026; Yang et al., 2024a; Kuang et al., 2025), often on top of depth foundation models (Piccinelli et al., 2024; 2025; Yang et al., 2024b;c; Lin et al., 2025; Bhat et al., 2023; Ke et al., 2024). These methods enforce temporal consistency with RGB-driven cues and learned priors and do not use sparse but metrically accurate sensor measurements. Test-time methods such as CVD (Luo et al., 2020) and RCVD (Kopf et al., 2021) fine-tune a network per video offline with estimated camera poses, whereas KFTP runs online with a frozen network. Kalman filtering has also been used to fuse per-pixel depth over time under known camera motion (Matthies et al., 1989; Hung & Ho, 1999), to update IMU-based ego-motion within unsupervised depth learning (Zhang et al., 2022), and for video restoration (Lu et al., 2018) and optical flow estimation (Bao et al., 2019). These methods filter depth, ego-motion, or the task output itself and therefore rely on a measurement model, camera motion, or inertial sensing, whereas KFTP filters the visual prompt of a frozen foundation model.

2.2 Depth Completion

Supervised approaches fuse RGB images and sparse depth on individual frames using paired dense ground truth (Bartolomei et al., 2024; Cheng et al., 2019; 2020; Huang et al., 2019; Imran et al., 2019; Jaritz et al., 2018; Jun et al., 2024; Ku et al., 2018; Lin et al., 2024; Liu et al., 2022b; Ma & Karaman, 2018; Ma et al., 2019; Park et al., 2020; Park & Jeon, 2024; Krishna & Vandrotti, 2023; Park et al., 2014), and unsupervised methods (Chancán et al., 2025; Chen et al., 2024; Liu et al., 2022a; Lopez-Rodriguez et al., 2020; Rim et al., 2025; Shivakumar et al., 2019; Wong &

Soatto, 2021; Wong et al., 2020; 2021a;b; Wu et al., 2024; Yan et al., 2023) reduce reliance on dense GT by leveraging photometric and geometric constraints. Both are typically specialized to the training distribution and degrade under out-of-domain shifts. Test-time adaptation methods such as Zero-DC (Hyoseok et al., 2025), Marigold-DC (Viola et al., 2025), and TestPromptDC (Jeong et al., 2025) adapt a frozen backbone to sparse measurements without retraining, but optimize each frame independently. Concurrent works (Kim et al., 2024; Zhu et al., 2025) incorporate temporal attention or fusion but can degrade under domain shifts, and CAPA (Ke et al., 2026) shares scene-level parameters across a sequence, which may be less suitable for streaming input with continuously changing motion, occlusions, and sensor reliability.

3 Background

3.1 Test-time Visual Prompting for Depth Completion

Modern foundation models for depth estimation learn strong geometric priors from large-scale datasets and produce depth estimates from images. To adapt such foundation models for depth completion, test-time visual prompting introduces a small set of learnable parameters without updating model weights (Jeong et al., 2025). Specifically, the pixel-wise visual prompt is added to the input images, and the prompt is then optimized to align dense predictions with sparse sensor measurements, enabling lightweight test-time adaptation while preserving pre-trained geometric knowledge. Formally, given an image I, frozen depth foundation model Φ, and a visual prompt V, we optimize V by minimizing the discrepancy between the predicted dense depth and sparse depth measurements:

$$V^* = \arg\min_V \sum_{i,j} M_{i,j} \left| \Phi(I+V)_{i,j} - S_{i,j} \right|, \tag{1}$$

where M denotes validity mask, S sparse sensor depth, V learnable prompt, and (i, j) indexes spatial pixel locations. At inference time, the optimized prompt V^* is added directly to the input image, forming a prompted image $I + V^*$. Consequently, the adapted dense depth map D is predicted as:

$$D = \Phi(I + V^*). \tag{2}$$

3.2 Kalman Filter Algorithm

The Kalman filter (Kalman, 1960) estimates a state by recursively alternating between a prediction step and an update step. A key advantage of the Kalman filter is to jointly estimate the state and its uncertainty, enabling uncertainty-aware fusion of sequential information.

Prediction Step. Given the previous state estimate $x_{t-1|t-1}$ and its covariance $P_{t-1|t-1}$, the filter computes the estimate at time t as:

$$x_{t|t-1} = F_t x_{t-1|t-1}, \tag{3}$$

$$P_{t|t-1} = F_t P_{t-1|t-1} F_t^\top + Q_t, \tag{4}$$

where F_t denotes the state transition matrix, and Q_t is the process noise covariance that accounts for uncertainty in the state transition. The pair $(x_{t|t-1}, P_{t|t-1})$ represents the predicted state before incorporating new observation.

Update Step. Given an observation z_t at time t, we model it as $z_t = H_t x_t + v_t$, where H_t is the observation matrix and $v_t \sim \mathcal{N}(0, R_t)$ denotes measurement noise. The Kalman gain is then computed as

$$K_t = P_{t|t-1} H_t^\top \left(H_t P_{t|t-1} H_t^\top + R_t \right)^{-1}. \tag{5}$$

Using the computed Kalman gain K_t and the observation matrix H_t, we update the prior estimate as,

$$x_{t|t} = x_{t|t-1} + K_t (z_t - H_t x_{t|t-1}), \tag{6}$$

$$P_{t|t} = (I - K_t H_t) P_{t|t-1}. \tag{7}$$

3.3 Visual Prompt Update via Kalman Filter Algorithm

In this work, we model the visual prompt V as a time-evolving state and estimate it via a Kalman filter. In the prediction step, we propagate the previous state estimate $(x_{t-1|t-1}, P_{t-1|t-1})$, which

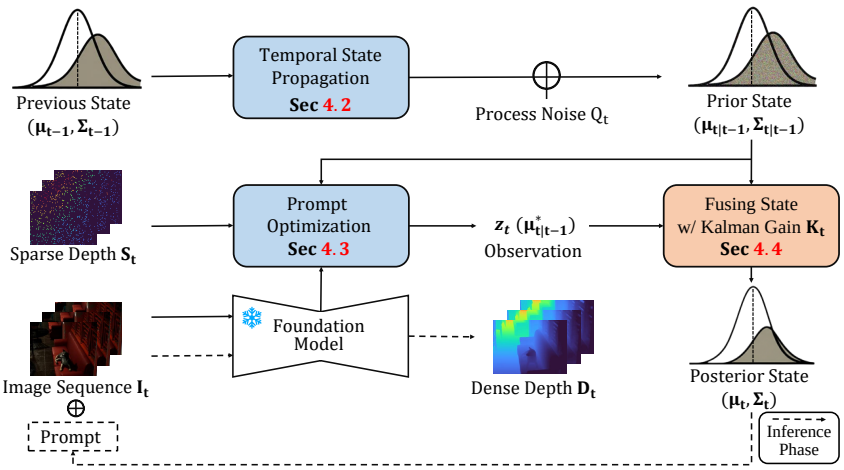


Figure 2: Overview of KFTP. We model the visual prompt as a temporal state $(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ for video depth completion.

represents a prompt for time $t-1$, to the current frame (Equations (3) and (4)). In the update step, we treat the optimized prompt V^* obtained by test-time optimization (Equation (1)) as a measurement z_t with an uncertainty map R_t. To account for the relative uncertainties of the propagated prior and the measurement, we then compute the Kalman gain K_t to fuse z_t with the propagated prior estimate $(x_{t|t-1}, P_{t|t-1})$ (Equations (5) to (7)). The computed posterior mean $x_{t|t}$ is then utilized as the prompt input to the frozen foundation model Φ to predict dense depth for frame t (Equation (2)).

4 Methodology

4.1 Problem Definition and Formulation for Temporal Prompting

Extending Visual Prompting to Video Sequences. Given a video sequence $\{\mathbf{I}_t\}_{t=1}^T$ and sparse depth measurements $\{\mathbf{S}_t\}_{t=1}^T$, our goal is to estimate dense, geometrically accurate, and temporally consistent depth maps $\mathbf{D}_t$. We adopt a pretrained monocular depth foundation model and keep its parameters frozen during inference. To enable lightweight online adaptation, we introduce a temporal prompt $\mathbf{V}_t \in \mathbb{R}^{C\times H\times W}$, where C denotes the prompt channel dimension and H, W denote the spatial height and width. We obtain a frame-wise prompted solution and corresponding dense depth map D_t by optimizing the prompt to match sparse measurements as follows:

$$\mathbf{V}_t^* = \arg\min_{\mathbf{V}_t} \sum_{i,j} \mathbf{M}_{t,i,j} \left|\Phi(\mathbf{I}_t + \mathbf{V}_t)_{i,j} - \mathbf{S}_{t,i,j}\right|, \tag{8}$$

$$\mathbf{D}_t = \Phi(\mathbf{I}_t + \mathbf{V}_t^*), \tag{9}$$

where, valid mask $\mathbf{M}_t \in \{0,1\}^{H\times W}$ indicates pixels where sparse measurements $\mathbf{S}_t$ are available, and (i, j) indexes spatial pixel locations.

Temporal Prompting for Kalman-Filter. To extend conventional visual prompt tuning (Jeong et al., 2025) to video, we reformulate the visual prompt as a time-evolving state that can be estimated sequentially. In other words, instead of optimizing an independent deterministic prompt per frame, we track a prompt state over time using a Kalman filter (Section 3.2). The state at frame t consists of a prompt estimate $\boldsymbol{\mu}_t$ and an element-wise prompt uncertainty map $\boldsymbol{\Sigma}_t$,

$$\boldsymbol{\mu}_t \in \mathbb{R}^{C\times H\times W}, \qquad \boldsymbol{\Sigma}_t \in \mathbb{R}^{C\times H\times W}_{>0}, \tag{10}$$

where a larger value of $\boldsymbol{\Sigma}_t$ indicates that the corresponding prompt element is less trusted. Maintaining a full uncertainty matrix over all $N = C \times H \times W$ elements is computationally intractable, so we use an element-wise map. This is a design choice for efficiency, not an assumption that the prompt elements are independent. On ten KITTI scenes, local 3×3 and 5×5 uncertainty blocks

change RMSE and MAE by at most 0.001 and 0.003 but increase run time by 10.4% and 32.7%. All operations on uncertainty maps are applied element-wise.

4.2 Prediction Step: Temporal State Propagation

We align the previous prompt estimate and its uncertainty map using backward optical flow (Dong & Fu, 2024) $\mathbf{F}_{t\to t-1}$ and a backward warping operator $\mathcal{W}(\cdot)$, which spatially aligns the previous state to the current frame using bilinear interpolation. The propagation of prior state $(\boldsymbol{\mu}_{t-1}, \boldsymbol{\Sigma}_{t-1})$ can be computed as below:

$$\boldsymbol{\mu}_{t|t-1} = \mathcal{W}(\boldsymbol{\mu}_{t-1}, \mathbf{F}_{t\to t-1}), \tag{11}$$

$$\boldsymbol{\Sigma}_{t|t-1} = \mathcal{W}(\boldsymbol{\Sigma}_{t-1}, \mathbf{F}_{t\to t-1}) + \mathbf{Q}_t, \tag{12}$$

where $\mathbf{Q}_t$ denotes the adaptive process noise due to warping errors which is formulated as:

$$\mathbf{Q}_t = \mathbf{Q}_t^{\text{base}} + \mathbf{Q}_t^{\text{feat}} + \mathbf{Q}_t^{\text{geo}} + \mathbf{Q}_t^{\text{motion}}, \tag{13}$$

Base term ($\mathbf{Q}_t^{\text{base}}$): This term indicates noise floor (Anderson, 2001) to avoid degenerate uncertainties (e.g., near-zero variances), ensuring stable Kalman gains and preventing overconfident updates.

Feature uncertainty ($\mathbf{Q}_t^{\text{feat}}$): We use channel-wise feature variation as an ambiguity cue and assign higher uncertainty to regions with low variation:

$$\mathbf{Q}_t^{\text{feat}} \propto \left(\text{std}_{\text{ch}}(\Phi_f(\mathbf{I}_t)) + \epsilon\right)^{-1}, \tag{14}$$

where $\text{std}_{\text{ch}}(\cdot)$ indicates the channel-wise standard deviation, and $\epsilon > 0$ is a constant for numerical stability.

Geometric error ($\mathbf{Q}_t^{\text{geo}}$): To detect occlusions or warping failures, we compute the forward-backward cycle consistency error:

$$\mathbf{Q}_t^{\text{geo}} = \left\|\mathbf{F}_{t\to t-1} + \mathcal{W}(\mathbf{F}_{t-1\to t}, \mathbf{F}_{t\to t-1})\right\|_2, \tag{15}$$

where $\mathbf{F}_{t-1\to t}$ is the forward flow, and $\|\cdot\|_2$ denotes the per-pixel ℓ_2 norm over the 2D flow vector. Large errors indicate unreliable propagation and are mapped to higher uncertainty.

Motion magnitude ($\mathbf{Q}_t^{\text{motion}}$): Large displacements in sequential frames easily induce flow errors and warping artifacts, so we quantify the uncertainty regarding motion as below:

$$\mathbf{Q}_t^{\text{motion}} \propto \|\mathbf{F}_{t\to t-1}\|_2. \tag{16}$$

Temporal Coherence of Prompts. For a fixed flow, bilinear warping is linear in the warped prompt (Jaderberg et al., 2015), so Equation (11) is a time-varying linear propagation that aligns the previous prompt to the current frame and carries it forward. Prompts optimized independently per frame already share such structure: after flow alignment, the cosine similarity of adjacent-frame prompts rises in the mid- and low-frequency Laplacian bands (Burt & Adelson, 1983) (e.g., $0.157 \to 0.279$ on KITTI) but hardly in the highest-frequency band ($0.059 \to 0.068$), so not every component propagates reliably.

4.3 Prediction Step: State Optimization for Observation $\mathbf{z}_t$

To perform the Kalman update, which combines the propagated prior and the current observation to produce a refined posterior state, we first construct a new observation $\mathbf{z}_t$ that incorporates information from the current frame. It is obtained by test-time optimization initialized from the propagated prior mean $\boldsymbol{\mu}_{t|t-1}$ (i.e., the propagated prompt), in which we optimize the prompt on the current frame $\mathbf{I}_t$ and sensor measurement $\mathbf{S}_t$, yielding an optimized prompt $\boldsymbol{\mu}^*_{t|t-1}$.

Optimization Objective. We obtain the new observation $\mathbf{z}_t$, under sparse depth supervision $\mathbf{S}_t$, by minimizing:

$$\mathcal{L}(\mu_{t|t-1}) = \mathcal{L}_{\text{sparse}} + \lambda_p \mathcal{L}_{\text{poisson}} + \lambda_t \mathcal{L}_{\text{temporal}}, \tag{17}$$

where $\mathbf{D}_t = \Phi(\mathbf{I}_t + \mathbf{V}_t)$ and λ_p, λ_t denote scale parameters. Each loss is described as follows:

Sparse Supervision ($\mathcal{L}_{sparse}$). We supervise valid pixels using the mask $\mathbf{M}_t$ derived from sparse depth $\mathbf{S}_t$. Following widely used practices (Jeong et al., 2025), we combine ℓ_1 and ℓ_2 losses to provide robust supervision:

$$\mathcal{L}_{\text{sparse}} = \sum_{i,j} \mathbf{M}_{t,i,j} \, |\mathbf{D}_{t,i,j} - \mathbf{S}_{t,i,j}| + \sqrt{\sum_{i,j} \mathbf{M}_{t,i,j} \, (\mathbf{D}_{t,i,j} - \mathbf{S}_{t,i,j})^2}, \tag{18}$$

Poisson-Based Regularization ($\mathcal{L}_{poisson}$). To encourage locally smooth and physically plausible surfaces, we impose second-order smoothness and an integrability constraint:

$$\mathcal{L}_{\text{poisson}} = \|\partial_{xx}\mathbf{D}_t\|_1 + \|\partial_{yy}\mathbf{D}_t\|_1 + \|\partial_y\partial_x\mathbf{D}_t - \partial_x\partial_y\mathbf{D}_t\|_1, \tag{19}$$

The first two terms suppress high-frequency artifacts and encourage locally smooth geometry. The third term enforces discrete integrability under finite differences and warping by encouraging commutativity of mixed partial derivatives during optimization.

Temporal Consistency ($\mathcal{L}_{temporal}$). We enforce soft temporal consistency in probability space. We interpret the predicted depth map as a per-pixel discrete probability distribution using Gaussian soft assignment (Yang et al., 2019) over K log-spaced depth bins $\{b_k\}_{k=1}^K$:

$$p_t^{(k)}(\mathbf{x}) = \frac{\exp\left(-\frac{(\log \mathbf{D}_t(\mathbf{x}) - \log b_k)^2}{2\sigma^2}\right)}{\sum_{j=1}^K \exp\left(-\frac{(\log \mathbf{D}_t(\mathbf{x}) - \log b_j)^2}{2\sigma^2}\right)}, \tag{20}$$

Here, $p_t^{(k)}(\mathbf{x})$ denotes the soft probability that pixel $\mathbf{x}$ in the frame at time t belongs to the k-th depth bin b_k. We warp the previous-frame distribution into the current frame:

$$p_{t|t-1} = \mathcal{W}(p_{t-1}, \mathbf{F}_{t\to t-1}), \tag{21}$$

and minimize the Kullback–Leibler divergence (Kullback & Leibler, 1951) averaged over spatial locations:

$$\mathcal{L}_{\text{temporal}} = \frac{1}{HW} \sum_{\mathbf{x}} \mathrm{KL}\big(p_t(\cdot|\mathbf{x}) \,\|\, p_{t|t-1}(\cdot|\mathbf{x})\big), \tag{22}$$

We scale the temporal term using a motion-adaptive weight λ_t, computed from the optical-flow magnitude (Equation (16)) and the cycle-consistency error (Equation (15)), which relaxes the constraint in regions with large motion or occlusion. The resulting $\mathbf{z}_t$ serves as a noisy observation of the prompt state, but, being initialized from the prior, it is not an independent linear measurement. Since $\mathbf{z}_t$ and $\boldsymbol{\mu}_{t|t-1}$ have the same shape, the observation matrix H_t of Equations (5) to (7) is the identity, and the nonlinearity of Φ enters only through Equation (28).

4.4 Update Step: State Fusion with Kalman Gain

Following the Kalman filter update (Section 3.2), we fuse the propagated prior $(\boldsymbol{\mu}_{t|t-1}, \boldsymbol{\Sigma}_{t|t-1})$ and the observation $\mathbf{z}_t$ to obtain the posterior state $(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$.d We obtain the posterior prompt as the minimizer of the per-frame weighted least-squares objective

$$J_t(\boldsymbol{\mu}) = \tfrac{1}{2}\|\boldsymbol{\mu} - \boldsymbol{\mu}_{t|t-1}\|^2_{\boldsymbol{\Sigma}^{-1}_{t|t-1}} + \tfrac{1}{2}\|\boldsymbol{\mu} - \mathbf{z}_t\|^2_{\mathbf{R}_t^{-1}}, \tag{23}$$

in which the inverse uncertainty maps act as fusion weights. Since the reliability of the observation varies across pixels, we model this spatially varying uncertainty using an element-wise observation uncertainty map $\mathbf{R}_t$, defined as:

$$\mathbf{R}_t = \mathbf{R}_\Phi + \beta \tanh(\mathbf{Q}_t^{\text{geo}}), \tag{24}$$

where $\mathbf{R}_\Phi$ captures the predictive uncertainty of the foundation model Φ, which is computed based on the uncertainty map of the foundation model (Piccinelli et al., 2025). Figure 3 illustrates how $\boldsymbol{\Sigma}_{t|t-1}$ and $\mathbf{R}_t$ determine the Kalman gain that weights the two estimates.

Table 1: Quantitative comparison on outdoor datasets (KITTI, DDAD, and vKITTIv2).

Model	Task	KITTI			DDAD			vKITTIv2		
		RMSE	MAE	TEPE	RMSE	MAE	TEPE	RMSE	MAE	TEPE
UniDepth V2 (Piccinelli et al., 2025)	Monocular Metric Depth Estimation	2.687	1.376	0.552	5.736	2.853	1.591	8.511	5.477	1.246
DepthAnything V3 (Lin et al., 2025)	Monocular Metric Depth Estimation	3.184	1.725	0.632	6.387	4.056	1.273	11.424	7.148	1.741
VideoDepthAnything (Chen et al., 2025)	Monocular Video Depth Estimation	3.639	2.054	0.470	10.423	6.909	**0.984**	14.581	12.413	**1.232**
DepthCrafter (Hu et al., 2025)	Monocular Video Depth Estimation	7.421	5.183	0.777	14.277	12.191	1.112	10.323	7.828	1.612
Zero-DC (Hyoseok et al., 2025)	Monocular Depth Completion	1.515	0.509	0.514	7.163	3.621	3.813	6.287	3.375	3.325
Marigold-DC (Viola et al., 2025)	Monocular Depth Completion	1.483	0.550	0.478	6.401	2.708	2.888	5.525	2.462	3.012
TestPromptDC (Jeong et al., 2025)	Monocular Depth Completion	1.480	0.482	0.404	5.652	1.963	1.946	4.776	1.717	2.058
KFTP (ours)	Video Depth Completion	**1.425**	**0.410**	**0.389**	**2.977**	**0.903**	1.057	**4.123**	**1.294**	1.425

Table 2: Quantitative comparison on indoor datasets (DyDToF, Sintel and NYUv2).

Model	Task	DyDToF			Sintel			NYUv2		
		RMSE	MAE	TEPE	RMSE	MAE	TEPE	RMSE	MAE	TEPE
UniDepth V2 (Piccinelli et al., 2025)	Monocular Metric Depth Estimation	5.014	3.294	0.639	4.255	2.938	1.414	0.219	0.140	0.140
DepthAnything V3 (Lin et al., 2025)	Monocular Metric Depth Estimation	3.029	2.046	0.802	13.677	10.732	4.049	0.229	0.129	0.097
VideoDepthAnything (Chen et al., 2025)	Monocular Video Depth Estimation	3.064	2.128	0.629	5.444	3.950	1.137	0.280	0.161	0.082
DepthCrafter (Hu et al., 2025)	Monocular Video Depth Estimation	3.906	2.348	0.668	14.277	12.191	1.112	0.623	0.484	0.112
Zero-DC (Hyoseok et al., 2025)	Monocular Depth Completion	3.399	2.038	1.490	6.591	1.799	1.734	0.152	0.097	0.099
Marigold-DC (Viola et al., 2025)	Monocular Depth Completion	2.662	1.094	1.455	5.580	1.159	1.599	0.135	0.053	0.072
TestPromptDC (Jeong et al., 2025)	Monocular Depth Completion	2.311	0.927	1.070	5.212	1.389	1.698	0.128	0.054	0.061
KFTP (ours)	Video Depth Completion	**0.940**	**0.295**	**0.373**	**1.577**	**0.540**	**0.574**	**0.126**	**0.052**	**0.060**

Kalman Update. Since all uncertainty maps are element-wise, the Kalman gain is also computed element-wise:

$$\mathbf{K}_t = \boldsymbol{\Sigma}_{t|t-1} \oslash \left(\boldsymbol{\Sigma}_{t|t-1} + \mathbf{R}_t\right), \tag{25}$$

where $\oslash$ denotes element-wise division. Pixels with large prior uncertainty and reliable observations yield $\mathbf{K}_t \approx 1$, whereas unreliable observations produce $\mathbf{K}_t \approx 0$. The posterior state is updated as:

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t \odot (\mathbf{z}_t - \boldsymbol{\mu}_{t|t-1}), \tag{26}$$

$$\boldsymbol{\Sigma}_t = (\mathbf{1} - \mathbf{K}_t) \odot \boldsymbol{\Sigma}_{t|t-1}, \tag{27}$$

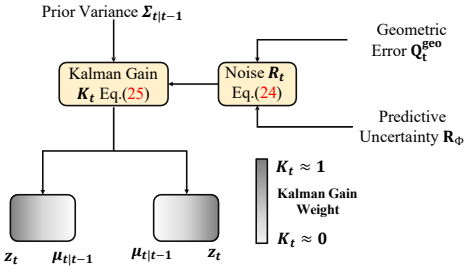


Figure 3: Uncertainty-aware Kalman update. The element-wise gain K_t adaptively balances the propagated prior and current observation according to their relative uncertainties.

where $\odot$ denotes element-wise multiplication and $\mathbf{1}$ is an all-ones tensor. We then infer the final dense depth map as:

$$\mathbf{D}_t = \Phi(\mathbf{I}_t + \boldsymbol{\mu}_t). \tag{28}$$

Setting $\nabla J_t = \mathbf{0}$ gives Equations (25) and (26), and the inverse Hessian $(\boldsymbol{\Sigma}_{t|t-1}^{-1} + \mathbf{R}_t^{-1})^{-1}$ gives Equation (27). Because the maps are element-wise, J_t separates per element, so the update is its exact minimizer and requires no Gaussian assumption. Since $\boldsymbol{\Sigma}_{t|t-1}$ and $\mathbf{R}_t$ are designed uncertainty maps rather than calibrated covariances, we do not claim the optimality of a probabilistic Kalman filter.

5 Experiments

This section evaluates our method under zero-shot test-time adaptation for video depth completion, where the depth foundation model remains frozen and only video frames and sparse depth measurements are used. Datasets, evaluation protocol, baselines, and metrics (RMSE/MAE/TEPE) are described in Appendix C.

5.1 Quantitative Comparison

Table 1 demonstrates that KFTP improves both spatial accuracy and temporal stability over TTA baselines on real-world and synthetic outdoor datasets (Guizilini et al., 2020; Cabon et al., 2020; Geiger et al., 2012). For indoor scenarios as shown in Table 2, KFTP demonstrates strong robustness under sparse supervision and challenging motion patterns. KFTP yields clear improvements on DyDToF (Sun et al., 2023) and Sintel (Butler et al., 2012), where non-rigid motion and depth discontinuities make frame-wise adaptation brittle. On NYUv2 (Silberman et al., 2012), KFTP also consistently improves over existing depth completion and foundation-model-based baselines.

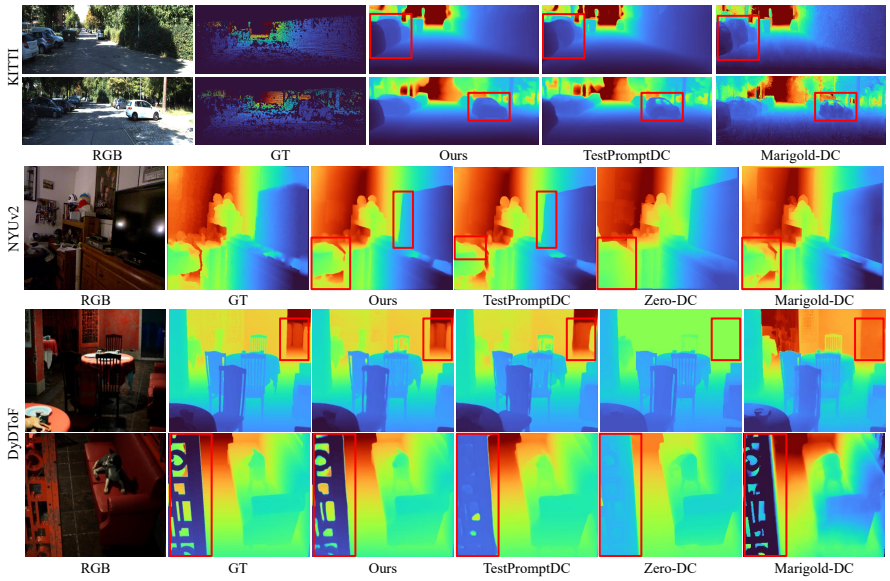


Figure 4: **Qualitative comparison.** Depth completion results on NYUv2 and KITTI. KFTP produces sharper geometry and more temporally stable depth than the comparison methods.

Table 3: Per-dataset comparison with CAPA (Ke et al., 2026). Δ is the relative change from frame-wise prompting (TestPromptDC), and a negative Δ is an improvement.

Dataset	TestPromptDC (Jeong et al., 2025)		CAPA (Ke et al., 2026)				Ours			
	RMSE ↓	MAE ↓	RMSE ↓	Δ	MAE ↓	Δ	RMSE ↓	Δ	MAE ↓	Δ
DDAD	5.652	1.963	3.869	−31.5%	1.410	−28.2%	**2.977**	−47.3%	**0.903**	−54.0%
KITTI	1.480	0.482	1.666	+12.6%	0.527	+9.3%	**1.425**	−3.7%	**0.410**	−14.9%
vKITTI	4.776	1.717	4.657	−2.5%	1.489	−13.3%	**4.123**	−13.7%	**1.294**	−24.6%
NYUv2	0.128	0.054	0.150	+17.2%	0.063	+16.7%	**0.126**	−1.6%	**0.052**	−3.7%
DyDToF	2.311	0.927	**0.707**	−69.4%	**0.198**	−78.6%	0.940	−59.3%	0.295	−68.2%
Sintel	5.212	1.389	1.823	−65.0%	**0.479**	−65.5%	**1.577**	−69.7%	0.540	−61.1%
Average	3.260	1.089	2.145	−34.2%	0.694	−36.2%	**1.861**	−42.9%	**0.582**	−46.5%

Table 4: Fusion rules with everything else fixed. None is frame-wise prompting, and EMA uses a constant gain $\mathbf{K}_t = \alpha \mathbf{1}$ in Equation (26).

Fusion rule	DDAD		KITTI		NYUv2	
	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓	RMSE ↓	MAE ↓
None (TestPromptDC)	5.652	1.963	1.480	0.482	0.128	0.054
EMA, $\alpha = 0.2$	3.272	1.227	1.469	0.515	0.136	0.059
EMA, $\alpha = 0.5$	3.308	1.257	1.472	0.517	0.135	0.059
EMA, $\alpha = 0.8$	3.242	1.203	1.470	0.515	0.136	0.059
Adaptive Fusion (Ours)	**2.977**	**0.903**	**1.425**	**0.410**	**0.126**	**0.052**

Dataset-wise gains. On KITTI and NYUv2, the frame-wise error of TestPromptDC is already much lower than on the other datasets, so little error remains to be recovered, and CAPA increases RMSE there by 12.6% and 17.2%, respectively (Table 3). Using temporal information therefore does not improve results automatically. On DDAD, frame-wise prompting barely improves the frozen UniDepthV2 backbone (-1.5%), and on Sintel it even degrades it ($+22.5\%$; Tables 1 and 2). Where the current-frame prompt is this unstable, a large recoverable error remains, and CAPA also reduces RMSE by 69.4%, 31.5%, and 65.0% on DyDToF, DDAD, and Sintel, so gains above 45% reflect this recoverable error rather than a behavior unique to KFTP. This difference is consistent with the adaptive gain: where the frame-wise prompt is already accurate, $\mathbf{K}_t$ can stay close to one and KFTP stays close to that result, so the change on KITTI and NYUv2 is small but still negative, and where it is unstable, $\mathbf{K}_t$ can decrease so that the propagated prior contributes more, and this is where the large improvements appear. A fixed weight cannot cover both regimes, and neither can CAPA.

5.2 Qualitative Comparison

Figure 4 qualitatively demonstrates that KFTP produces sharper and more stable depth maps across both outdoor and indoor scenarios. On KITTI, our method better preserves vehicle boundaries and distant scene structures under sparse depth supervision. On NYUv2, KFTP also recovers cleaner indoor geometry around furniture, walls, and object boundaries, where depth discontinuities and noisy measurements often degrade frame-wise baselines. Compared with TestPromptDC, Zero-DC, and Marigold-DC, KFTP reduces spurious artifacts in the highlighted regions and maintains more coherent depth structures. Additional video results are provided in the supplementary materials to further demonstrate temporal consistency.

5.3 Ablation Study

Table 4 separates the effect of temporal prompt fusion from that of the adaptive gain. On DDAD, where frame-wise prompting leaves a large error, fusing prompts over time already lowers RMSE from 5.652 to 3.242–3.308 even with a fixed weight, and the adaptive gain further reduces RMSE and MAE by 8.2% and 24.9% relative to the best fixed α. On KITTI and NYUv2, fixed blending degrades the frame-wise result (e.g., MAE $0.482 \rightarrow 0.515$ on KITTI), and the best α differs across datasets, whereas the adaptive gain improves on all three datasets without choosing a weight per dataset.

Effect of Adaptive Fusion Gain. The fusion gain determines how much the final prompt trusts the prompt propagated from the previous frame versus the observation fitted to the current sparse measurements. EMA fixes this trade-off with a single α for every element and frame, so its gain map is uniform (Figure 5), and every α leaves a large error in the highlighted region. The Kalman gain instead sets the trade-off per element from the relative uncertainty of the two (Equation (25)), trusting the current observation where propagation is unreliable and the propagated prior where the observation is unreliable, and largely reduces this error.

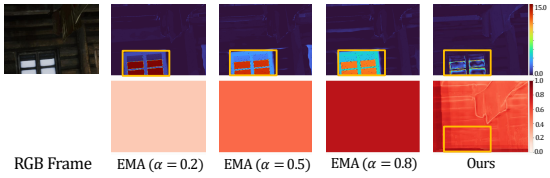


Figure 5: Qualitative analysis of EMA vs.Adaptive Fusion. Top: error maps. Bottom: $\mathbf{K}_t$ averaged over prompt channels (0: propagated prior, 1: current observation). Every fixed α leaves a large error in the highlighted region, which the Kalman gain reduces.

Temporal Stability. We further analyze frame-wise MAE over 40 consecutive frames in a single DyDToF sequence to directly assess temporal stability beyond averaged metrics. As shown in Fig. 6, our method produces a markedly smoother error trajectory with reduced frame-to-frame fluctuations compared to prior test-time adaptation baselines. In contrast, frame-wise optimization methods exhibit larger oscillations as motion and visibility change, reflecting sensitivity to sparse and noisy supervision. This indicates that Kalman filter-based temporal prompting stabilizes per-frame updates by propagating a temporally consistent prior and fusing it with the current observation, thereby mitigating flicker and improving temporal consistency.

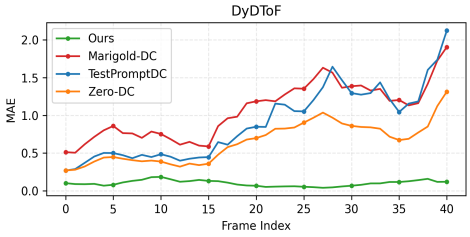


Figure 6: Frame-wise error over 40 frames. KFTP shows more stable error curves than prior methods, indicating improved temporal consistency.

6 Conclusion

We introduce Kalman Filter-based Temporal Prompting, a lightweight test-time adaptation framework for video depth completion on top of a frozen monocular depth foundation model. KFTP models the visual prompt as a time-evolving latent state, propagates it with motion guidance, and performs uncertainty-aware Kalman updates with adaptive process noise to remain robust under motion and occlusions. Experiments on indoor and outdoor benchmarks show that KFTP improves both spatial accuracy and temporal stability over frame-wise prompt tuning and improves average spatial accuracy over multi-frame baselines. KFTP does not use camera poses or multi-view geometry, and incorporating them could improve propagation in rigid regions. Future work will reduce test-time optimization overhead and further improve long-sequence propagation stability.

7 AI Disclosure

Generative AI served only as a writing aid in preparing this manuscript, where it was used to proofread the text and refine its phrasing. It played no role in designing the method, conducting the experiments, or interpreting the results. The authors checked all suggested changes and are accountable for the entire content of the paper.

REFERENCES

Jeffrey L Anderson. An ensemble adjustment kalman filter for data assimilation. *Monthly weather review*, 129 (12):2884–2903, 2001.

Wenbo Bao, Xiaoyun Zhang, Li Chen, Lianghui Ding, and Zhiyong Gao. KalmanFlow 2.0: Efficient video optical flow estimation via context-aware Kalman filtering. *IEEE Transactions on Image Processing*, 2019.

Luca Bartolomei, Matteo Poggi, Andrea Conti, Fabio Tosi, and Stefano Mattoccia. Revisiting Depth Completion from a Stereo Matching Perspective for Cross-Domain Generalization. In *3DV*, 2024.

Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Muller. ZoeDepth: Zero-Shot Transfer by Combining Relative and Metric Depth. *arXiv preprint arXiv:2302.12288*, 2023.

Peter J. Burt and Edward H. Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, 1983.

D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.) (ed.), *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pp. 611–625. Springer-Verlag, October 2012.

Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual KITTI 2. *arXiv preprint arXiv:2001.10773*, 2020.

Marvin Chancán, Alex Wong, and Ian Abraham. 3d reprojection-driven robot navigation improves depth sensing. In *2025 International Conference on Advanced Robotics and Mechatronics (ICARM)*, pp. 383–390. IEEE, 2025.

Sili Chen, Hengkai Guo, Shengnan Zhu, Feihu Zhang, Zilong Huang, Jiashi Feng, and Bingyi Kang. Video depth anything: Consistent depth estimation for super-long videos. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22831–22840, 2025.

Xien Chen, Rit Gangopadhyay, Michael Chu, Patrick Rim, Hyoungseob Park, and Alex Wong. Uncle: Benchmarking unsupervised continual learning for depth completion. *arXiv preprint arXiv:2410.18074*, 2024.

Xinjing Cheng, Peng Wang, and Ruigang Yang. Learning depth with convolutional spatial propagation network. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2361–2379, 2019.

Xinjing Cheng, Peng Wang, Chenye Guan, and Ruigang Yang. Cspn++: Learning context and resource aware convolutional spatial propagation networks for depth completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 10615–10622, 2020.

Gene Chou, Wenqi Xian, Guandao Yang, Mohamed Abdelfattah, Bharath Hariharan, Noah Snavely, Ning Yu, and Paul Debevec. Flashdepth: Real-time streaming video depth estimation at 2k resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9638–9648, 2025.

Qiaole Dong and Yanwei Fu. Memflow: Optical flow estimation and prediction with memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19068–19078, 2024.

Xingshuai Dong, Matthew A Garratt, Sreenatha G Anavatti, and Hussein A Abbass. Towards real-time monocular depth estimation for robotics: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(10): 16940–16961, 2022.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are We Ready for Autonomous Driving? the KITTI Vision Benchmark Suite. In *CVPR*, 2012.

Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *CVPR*, 2020.

Aleksander Holynski and Johannes Kopf. Fast depth densification for occlusion-aware augmented reality. *ACM Transactions on Graphics (ToG)*, 37(6):1–11, 2018.

Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. Depthcrafter: Generating Consistent Long Depth Sequences for Open-World Videos. In *CVPR*, 2025.

Zixuan Huang, Junming Fan, Shenggan Cheng, Shuai Yi, Xiaogang Wang, and Hongsheng Li. Hms-net: Hierarchical multi-scale sparsity-invariant network for sparse depth completion. *IEEE Transactions on Image Processing*, 29:3429–3441, 2019.

Jeffrey Humpherys, Preston Redd, and Jeremy West. A fresh look at the Kalman filter. *SIAM Review*, 54(4): 801–823, 2012.

Y. S. Hung and H. T. Ho. A Kalman filter approach to direct depth estimation incorporating surface structure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(6):570–575, 1999.

Lee Hyoseok, Kyeong Seon Kim, Kwon Byung-Ki, and Tae-Hyun Oh. Zero-shot depth completion via test-time alignment with affine-invariant depth prior. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 3877–3885, 2025.

Saif Muhammad Imran, Yunfei Long, Xiaoming Liu, and Daniel Morris. Depth Coefficients for Depth Completion. In *CVPR*, 2019.

Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *NeurIPS*, 2015.

Maximilian Jaritz, Raoul de Charette, Emilie Wirbel, Xavier Perrotton, and Fawzi Nashashibi. Sparse and Dense Data with CNNs: Depth Completion and Semantic Segmentation. In *3DV*, 2018.

Chanhwi Jeong, Inhwan Bae, Jin-Hwi Park, and Hae-Gon Jeon. Test-time prompt tuning for zero-shot depth completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9443–9454, 2025.

Jinyoung Jun, Jae-Han Lee, and Chang-Su Kim. Masked Spatial Propagation Network for Sparsity-Adaptive Depth Refinement. In *CVPR*, 2024.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9492–9502, 2024.

Bingxin Ke, Qunjie Zhou, Jiahui Huang, Xuanchi Ren, Tianchang Shen, Konrad Schindler, Laura Leal-Taixé, and Shengyu Huang. Depth completion as parameter-efficient test-time adaptation. *arXiv preprint arXiv:2602.14751*, 2026.

Jungeon Kim, Soongjin Kim, Jaesik Park, and Seungyong Lee. Deep cost ray fusion for sparse depth video completion. In *European Conference on Computer Vision*, pp. 328–346. Springer, 2024.

Johannes Kopf, Xuejian Rong, and Jia-Bin Huang. Robust consistent video depth estimation. In *CVPR*, 2021.

Sriram Krishna and Basavaraja Shanthappa Vandrotti. Deepsmooth: efficient and smooth depth completion. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 3358–3367. IEEE, 2023.

Jason Ku, Ali Harakeh, and Steven L. Waslander. In Defense of Classical Image Processing: Fast Depth Completion on the CPU. In *CRV*, 2018.

Zhengfei Kuang, Tianyuan Zhang, Kai Zhang, Hao Tan, Sai Bi, Yiwei Hu, Zexiang Xu, Milos Hasan, Gordon Wetzstein, and Fujun Luan. Buffer anytime: Zero-shot video depth and normal from image priors. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17660–17670, 2025.

Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Yuankai Lin, Hua Yang, Tao Cheng, Wending Zhou, and Zhouping Yin. DySPN: Learning Dynamic Affinity for Image-Guided Depth Completion. *IEEE TCSVT*, 2024.

Tian Yu Liu, Parth Agrawal, Allison Chen, Byung-Woo Hong, and Alex Wong. Monitored distillation for positive congruent depth completion. In *European Conference on Computer Vision*, pp. 35–53. Springer, 2022a.

Xin Liu, Xiaofei Shao, Bo Wang, Yali Li, and Shengjin Wang. Graphcspn: Geometry-aware depth completion via dynamic gcns. In *European conference on computer vision*, pp. 90–107. Springer, 2022b.

Adrian Lopez-Rodriguez, Benjamin Busam, and Krystian Mikolajczyk. Project to adapt: Domain adaptation for depth completion from noisy and sparse sensor data. In *Proceedings of the asian conference on computer vision*, 2020.

Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Zhiyong Gao, and Ming-Ting Sun. Deep Kalman filtering network for video compression artifact reduction. In *ECCV*, 2018.

Xuan Luo, Jia-Bin Huang, Richard Szeliski, Kevin Matzen, and Johannes Kopf. Consistent video depth estimation. *ACM Transactions on Graphics (ToG)*, 39(4):71–1, 2020.

Fangchang Ma and Sertac Karaman. Sparse-to-Dense: Depth Prediction from Sparse Depth Samples and a Single Image. In *ICRA*, 2018.

Fangchang Ma, Guilherme Venturelli Cavalheiro, and Sertac Karaman. Self-Supervised Sparse-to-Dense: Self-Supervised Depth Completion from LiDAR and Monocular Camera. In *ICRA*, 2019.

Larry Matthies, Takeo Kanade, and Richard Szeliski. Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, 3(3):209–238, 1989.

Zainab Oufqir, Abdellatif El Abderrahmani, and Khalid Satori. Arkit and arcore in serve to augmented reality. In *2020 International Conference on Intelligent Systems and Computer Vision (ISCV)*, pp. 1–7. IEEE, 2020.

Dennis Park, Rares Ambrus, Vitor Guizilini, Jie Li, and Adrien Gaidon. Is pseudo-lidar needed for monocular 3d object detection? In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3142–3152, 2021.

Jaesik Park, Hyeongwoo Kim, Yu-Wing Tai, Michael S Brown, and In So Kweon. High-quality depth map upsampling and completion for rgb-d cameras. *IEEE Transactions on Image Processing*, 23(12):5559–5572, 2014.

Jin-Hwi Park and Hae-Gon Jeon. A Simple yet Universal Framework for Depth Completion. In *NeurIPS*, 2024.

Jinsun Park, Kyungdon Joo, Zhe Hu, Chi-Kuei Liu, and In So Kweon. Non-local spatial propagation network for depth completion. In *European conference on computer vision*, pp. 120–136. Springer, 2020.

Vaishakh Patil, Wouter Van Gansbeke, Dengxin Dai, and Luc Van Gool. Don't forget the past: Recurrent depth estimation from monocular video. *IEEE Robotics and Automation Letters*, 5(4):6813–6820, 2020.

Luigi Piccinelli, Yung-Hsu Yang, Christos Sakaridis, Mattia Segu, Siyuan Li, Luc Van Gool, and Fisher Yu. Unidepth: Universal monocular metric depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10106–10116, 2024.

Luigi Piccinelli, Christos Sakaridis, Yung-Hsu Yang, Mattia Segu, Siyuan Li, Wim Abbeloos, and Luc Van Gool. Unidepthv2: Universal monocular metric depth estimation made simpler. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Luigi Piccinelli, Thiemo Wandel, Christos Sakaridis, Wim Abbeloos, and Luc Van Gool. Video depth propagation. In *3DV*, 2026.

Patrick Rim, Hyoungseob Park, Suchisrit Gangopadhyay, Ziyao Zeng, Younjoon Chung, and Alex Wong. Protodepth: Unsupervised continual depth completion with prototypes. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6304–6316. IEEE, 2025.

Jiahao Shao, Yuanbo Yang, Hongyu Zhou, Youmin Zhang, Yujun Shen, Vitor Guizilini, Yue Wang, Matteo Poggi, and Yiyi Liao. Learning temporally consistent video depth from video diffusion priors. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22841–22852. IEEE, 2025.

Shreyas S Shivakumar, Ty Nguyen, Ian D Miller, Steven W Chen, Vijay Kumar, and Camillo J Taylor. Dfusenet: Deep fusion of rgb and sparse depth information for image guided dense depth completion. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 13–20. IEEE, 2019.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In *ECCV*, 2012.

H. W. Sorenson. Least-squares estimation: From Gauss to Kalman. *IEEE Spectrum*, 7(7):63–68, 1970.

Zhanghao Sun, Wei Ye, Jinhui Xiong, Gyeongmin Choe, Jialiang Wang, Shuochen Su, and Rakesh Ranjan. Consistent direct time-of-flight video depth super-resolution. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pp. 5075–5085, 2023.

Jeffrey Uhlmann and Simon Julier. Gaussianity and the kalman filter: A simple yet complicated relationship. *arXiv preprint arXiv:2405.00058*, 2024.

Massimiliano Viola, Kevin Qu, Nando Metzger, Bingxin Ke, Alexander Becker, Konrad Schindler, and Anton Obukhov. Marigold-dc: Zero-shot monocular depth completion with guided diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5359–5370, 2025.

Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8445–8453, 2019.

Alex Wong and Stefano Soatto. Unsupervised depth completion with calibrated backprojection layers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 12747–12756, 2021.

Alex Wong, Xiaohan Fei, Stephanie Tsuei, and Stefano Soatto. Unsupervised depth completion from visual inertial odometry. *IEEE Robotics and Automation Letters*, 5(2):1899–1906, 2020.

Alex Wong, Safa Cicek, and Stefano Soatto. Learning topology from synthetic data for unsupervised depth completion. *IEEE Robotics and Automation Letters*, 6(2):1495–1502, 2021a.

Alex Wong, Xiaohan Fei, Byung-Woo Hong, and Stefano Soatto. An adaptive framework for learning unsupervised depth completion. *IEEE Robotics and Automation Letters*, 6(2):3120–3127, 2021b.

Yangchao Wu, Tian Yu Liu, Hyoungseob Park, Stefano Soatto, Dong Lao, and Alex Wong. Augundo: Scaling up augmentations for monocular depth completion and estimation. In *European Conference on Computer Vision*, pp. 274–293. Springer, 2024.

Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, and Dacheng Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8121–8130, 2022.

Zhiqiang Yan, Kun Wang, Xiang Li, Zhenyu Zhang, Jun Li, and Jian Yang. Desnet: Decomposed scale-consistent network for unsupervised depth completion. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 3109–3117, 2023.

Gengshan Yang, Peiyun Hu, and Deva Ramanan. Inferring distributions over depth from a single image. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 6090–6096. IEEE, 2019.

Honghui Yang, Di Huang, Wei Yin, Chunhua Shen, Haifeng Liu, Xiaofei He, Binbin Lin, Wanli Ouyang, and Tong He. Depth any video with scalable synthetic data. *arXiv preprint arXiv:2410.10815*, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything: Unleashing the Power of Large-Scale Unlabeled Data. In *CVPR*, 2024b.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth Anything V2. In *NeurIPS*, 2024c.

Haokui Zhang, Chunhua Shen, Ying Li, Yuanzhouhan Cao, Yu Liu, and Youliang Yan. Exploiting temporal consistency for real-time video depth estimation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1725–1734, 2019.

Sen Zhang, Jing Zhang, and Dacheng Tao. Towards scale-aware, robust, and generalizable unsupervised monocular depth estimation by integrating IMU motion dynamics. In *ECCV*, pp. 143–160, 2022.

Zheng Zhang, Lihe Yang, Tianyu Yang, Chaohui Yu, Xiaoyang Guo, Yixing Lao, and Hengshuang Zhao. Stabledepth: Scene-consistent and scale-invariant monocular depth. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7069–7078, 2025.

Zhoutong Zhang, Forrester Cole, Richard Tucker, William T Freeman, and Tali Dekel. Consistent depth of moving objects in video. *ACM Transactions on Graphics (ToG)*, 40(4):1–12, 2021.

Xuan Zhu, Jijun Xiang, Xianqi Wang, Longliang Liu, Yu Wang, Hong Zhang, Fei Guo, and Xin Yang. Svdc: Consistent direct time-of-flight video depth completion with frequency selective fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16619–16628, 2025.

A SUPPLEMENTARY VIDEO

We include a supplementary video to present additional results in dynamic scenes. Compared with static visualizations, the video more clearly illustrates the temporal coherence of the predictions, showing that the proposed method maintains better temporal consistency than existing approaches.

B ALGORITHM

Algorithm 1 KFTP for Video Depth Completion

Require: Video frames $\{\mathbf{I}_t\}_{t=1}^{T}$, sparse depth $\{\mathbf{S}_t\}_{t=1}^{T}$, frozen depth foundation model Φ, warping $\mathcal{W}(\cdot,\cdot)$, hyperparameters $(\lambda_p, \beta, K, \mathcal{H})$
Ensure: Dense depth maps $\{\mathbf{D}_t\}_{t=1}^{T}$

Initialize prompt posterior $(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$
for $t = 1$ **to** T **do**
  (1) Temporal state propagation
  Compute adaptive process noise: $\mathbf{Q}_t \leftarrow \mathbf{Q}_t^{\text{base}} + \mathbf{Q}_t^{\text{feat}} + \mathbf{Q}_t^{\text{geo}} + \mathbf{Q}_t^{\text{motion}}$
  Propagate: $\boldsymbol{\mu}_{t|t-1} \leftarrow \mathcal{W}(\boldsymbol{\mu}_{t-1}, \mathbf{F}_{t\to t-1})$, $\boldsymbol{\Sigma}_{t|t-1} \leftarrow \mathcal{W}(\boldsymbol{\Sigma}_{t-1}, \mathbf{F}_{t\to t-1}) + \mathbf{Q}_t$
  (2) Observation construction via test-time optimization
  Initialize prompt $\mathbf{V} \leftarrow \boldsymbol{\mu}_{t|t-1}$
  repeat
    Predict depth $\mathbf{D}_t \leftarrow \Phi(\mathbf{I}_t + \mathbf{V})$
    Compute $\mathcal{L} \leftarrow \mathcal{L}_{\text{sparse}} + \lambda_p \mathcal{L}_{\text{poisson}} + \lambda_t \mathcal{L}_{\text{temporal}}$
    Update $\mathbf{V}$ by gradient descent on $\mathcal{L}$
  until $\mathcal{H}$ steps are completed
  Set observation $\mathbf{z}_t \leftarrow \mathbf{V}$
  (3) Measurement noise modeling
  Compute predictive uncertainty $\mathbf{R}_\Phi$
  Compute geometric error $\mathbf{Q}_t^{\text{geo}}$
  Set $\mathbf{R}_t \leftarrow \mathbf{R}_\Phi + \beta \tanh(\mathbf{Q}_t^{\text{geo}})$
  (4) Kalman update
  Kalman gain: $\mathbf{K}_t \leftarrow \boldsymbol{\Sigma}_{t|t-1} \oslash (\boldsymbol{\Sigma}_{t|t-1} + \mathbf{R}_t)$
  Posterior State: $\boldsymbol{\mu}_t \leftarrow \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t \odot (\mathbf{z}_t - \boldsymbol{\mu}_{t|t-1})$, $\boldsymbol{\Sigma}_t \leftarrow (\mathbf{1} - \mathbf{K}_t) \odot \boldsymbol{\Sigma}_{t|t-1}$
  (5) Depth prediction
  $\mathbf{D}_t \leftarrow \Phi(\mathbf{I}_t + \boldsymbol{\mu}_t)$
end for
return $\{\mathbf{D}_t\}_{t=1}^{T}$

C EXPERIMENTAL SETUP

We evaluate KFTP on six benchmarks that cover real-world outdoor driving, indoor dynamics, and synthetic stress tests. Specifically, we use KITTI (Geiger et al., 2012), DDAD (Guizilini et al., 2020) and VKITTIv2 (Cabon et al., 2020) to assess outdoor performance. For indoor and highly dynamic settings, we evaluate on DyDToF (Sun et al., 2023) and NYUv2 (Silberman et al., 2012), and further use Sintel (Butler et al., 2012) as a challenging synthetic benchmark for temporal robustness under large motion and severe occlusions.

Outdoor Datasets. KITTI provides RGB frames with LiDAR-based sparse depth in real driving environments. DDAD contains long, high-resolution driving sequences with dense LiDAR measurements and complex motion patterns, making it well suited for evaluating temporal stability over extended horizons. VKITTIv2 provides synthetic driving sequences with dense ground truth depth, allowing for comprehensive evaluation in diverse outdoor scenarios.

Indoor Datasets. DyDToF includes challenging indoor sequences with non-rigid motion and depth discontinuities. NYUv2 contains a diverse set of real-world indoor scenes captured with an RGB-D camera, offering realistic complexities in everyday environments. Sintel provides synthetic sequences with large motion and severe occlusions and is widely used to stress-test temporal robustness. Across all datasets, the input consists of RGB video frames and their corresponding sparse depth maps. To ensure a strict zero-shot protocol, none of the evaluation datasets are used to train the underlying depth foundation model.

Comparison Methods. We compare KFTP against seven representative baselines spanning monocular depth estimation, video depth estimation, and test-time adaptation for depth completion. For RGB-only monocular metric depth estimation, we include UniDepth V2 (Piccinelli et al., 2025) and DepthAnything V3 (Lin et al., 2025), which predict metric depth from a single image without using sparse measurements.

For monocular video depth estimation, we compare with VideoDepthAnything (Chen et al., 2025) and DepthCrafter (Hu et al., 2025), which leverage multi-frame inputs or temporal modules to improve temporal consistency but still do not explicitly fuse sparse depth, providing a direct reference for video-based temporal modeling alone. Finally, for TTA-based depth completion, we evaluate Zero-DC (Hyoseok et al., 2025), Marigold-DC (Viola et al., 2025), and TestPromptDC (Jeong et al., 2025), which adapt pretrained depth models at inference time using sparse depth supervision while keeping the backbone frozen.

Evaluation Metrics. We evaluate both spatial accuracy and temporal consistency. Spatial metrics include RMSE and MAE, which measure per-frame depth. Given prediction $\hat{\mathbf{D}}_t$, ground truth $\mathbf{D}_t$, and valid mask $\mathbf{M}_t$ at time t:

$$\mathrm{RMSE} = \sqrt{\frac{\sum_t \sum_{\mathbf{i},\mathbf{j}} \mathbf{M}_t(\mathbf{i},\mathbf{j}) \left(\hat{\mathbf{D}}_t(\mathbf{i},\mathbf{j}) - \mathbf{D}_t(\mathbf{i},\mathbf{j})\right)^2}{\sum_t \sum_{\mathbf{i},\mathbf{j}} \mathbf{M}_t(\mathbf{i},\mathbf{j})}}, \tag{29}$$

$$\mathrm{MAE} = \frac{\sum_t \sum_{\mathbf{i},\mathbf{j}} \mathbf{M}_t(\mathbf{i},\mathbf{j}) \left|\hat{\mathbf{D}}_t(\mathbf{i},\mathbf{j}) - \mathbf{D}_t(\mathbf{i},\mathbf{j})\right|}{\sum_t \sum_{\mathbf{i},\mathbf{j}} \mathbf{M}_t(\mathbf{i},\mathbf{j})}, \tag{30}$$

For temporal consistency, using optical-flow warping (Xu et al., 2022) $\mathcal{W}(\cdot, \mathbf{F}_{t\to t+1})$:

$$\mathrm{TEPE} = \frac{1}{T-1} \sum_{t=1}^{T-1} \frac{\sum_{\mathbf{i},\mathbf{j}} \left|\mathcal{W}(\hat{\mathbf{D}}_t, \mathbf{F}_{t\to t+1})(\mathbf{i},\mathbf{j}) - \hat{\mathbf{D}}_{t+1}(\mathbf{i},\mathbf{j})\right|}{HW}. \tag{31}$$

Implementation Details. We implement our method in PyTorch and conduct all experiments on a single NVIDIA B200 GPU. For a fair comparison with the previous state-of-the-art model, TestPromptDC (Jeong et al., 2025), we adopt the same pre-trained UniDepthV2 (Piccinelli et al., 2025) as our frozen monocular depth foundation backbone. The visual prompt is injected additively into the input space, and optical flow is estimated using a pre-trained flow network (Dong & Fu, 2024). The uncertainty maps are element-wise, and the update follows Eqs. (25)–(27). During inference, we perform short-run test-time optimization per frame. Specifically, we utilize the AdamW optimizer together with the OneCycleLR scheduler, starting with an initial learning rate of $2e{-}3$. Unless otherwise specified, we perform 50 test-time optimization steps per frame, where the sparse loss uses a weight of 1.0 and both λ_p and λ_t are set to 0.1.

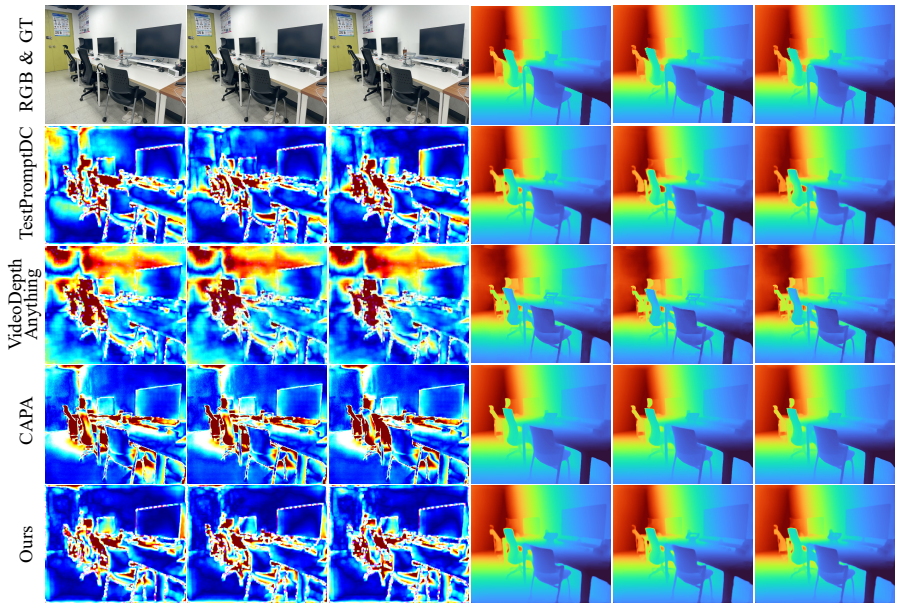


Figure 7: Qualitative comparison on real-world sequences captured with an iPhone 16 Pro using Apple ARKit (Oufqir et al., 2020). Compared with prior depth completion methods, the proposed method produces more complete and temporally consistent depth maps.

Table 5: Quantitative comparison on real-world iPhone sequences captured with Apple ARKit.

Model	RMSE ↓	MAE ↓	TEPE ↓
VideoDepthAnything Chen et al. (2025)	0.5838	0.3059	0.0915
TestPromptDC Jeong et al. (2025)	0.1850	0.0589	0.0643
CAPA Ke et al. (2026)	0.3358	0.1143	0.1065
Ours	**0.1770**	**0.0564**	**0.0570**

D Real-World Dataset: Apple iPhone

We further evaluate KFTP on real-world sequences captured with an iPhone 16 Pro using Apple ARKit (Oufqir et al., 2020). Unlike curated benchmark datasets, these sequences contain practical challenges such as handheld camera motion, sensor noise, incomplete sparse depth observations, textureless indoor regions, and depth discontinuities around furniture and object boundaries.

As shown in Figure 7, KFTP produces more complete and temporally coherent depth maps than existing baselines. While VideoDepthAnything, TestPromptDC, and CAPA often exhibit noisy local structures or unstable depth responses across frames, our method preserves cleaner object boundaries and more consistent scene geometry. This improvement comes from temporal prompt propagation and uncertainty-aware updates, which reduce sensitivity to noisy or insufficient sparse observations.

Table 5 quantitatively confirms this trend. KFTP achieves the best RMSE, MAE, and TEPE, indicating improvements in both spatial accuracy and temporal stability. These results demonstrate that KFTP generalizes well to real-world mobile AR scenarios and can robustly adapt to sparse depth inputs from practical sensing devices.

E Robustness against Depth Sparsity

We further evaluate robustness under *severe* depth sparsity by downsampling the input sparse depth measurements across both outdoor and indoor datasets, including KITTI (64→8 lines), DyDToF (500→50), NYUv2 (500→50), Sintel (500→50), and VKITTIv2 (1500→150). As shown in Tab. 6, KFTP consistently outperforms the strongest per-frame prompt baseline, TestPromptDC (Jeong et al., 2025), across all datasets and metrics. The gains are observed not only in spatial accuracy, measured

Table 6: Quantitative comparison under sparse depth settings. Lower is better for all metrics.

Model	KITTI (64 → 8-line)			DyDToF (500 → 50)			NYUv2 (500 → 50)			Sintel (500 → 50)			VKITTIv2 (1500 → 150)		
	RMSE ↓	MAE ↓	TEPE ↓	RMSE ↓	MAE ↓	TEPE ↓	RMSE ↓	MAE ↓	TEPE ↓	RMSE ↓	MAE ↓	TEPE ↓	RMSE ↓	MAE ↓	TEPE ↓
TestPromptDC Jeong et al. (2025)	2.3177	0.6995	0.6561	2.9598	1.0703	1.2105	0.1657	0.0734	0.0876	5.8216	1.6621	1.8780	5.2645	2.3125	2.2415
Ours	**1.9432**	**0.5687**	**0.5191**	**1.2839**	**0.4201**	**0.5349**	**0.1562**	**0.0723**	**0.0856**	**1.9364**	**0.8975**	**0.8842**	**4.9189**	**2.1553**	**2.0124**

by RMSE and MAE, but also in TEPE, indicating improved temporal stability under extremely sparse observations.

These results show that the proposed temporal prompt propagation and uncertainty-aware fusion remain effective even when current-frame depth measurements are highly limited. In such low-measurement regimes, frame-wise adaptation can easily overfit to sparse or noisy observations, leading to unstable depth predictions across frames. In contrast, KFTP leverages temporally propagated prompt states as reliable priors, enabling more stable and accurate depth completion under severe sparsity.

F Computational Cost Analysis

Table 7: Quantitative comparison of model efficiency on KITTI-DC.

Models	Required Memory (MB) ↓	Inference Speed (s/image) ↓	FLOPs	Params (M)
VideoDepthAnything Chen et al. (2025)	3242.7	0.27	234.20	384.39
SVDC Zhu et al. (2025)	636.8	1.14	298.19	24.34
Zero-DC Hyoseok et al. (2025)	8727	20.56	-	-
Marigold-DC Viola et al. (2025)	11372	21.67	-	-
TestPromptDC Jeong et al. (2025)	2885	5.33	187.05	363.21
CAPA Ke et al. (2026)	30366	0.75	41.09	1256.93
Ours	3008	5.39	189.02	363.21

We compare the efficiency of different methods on KITTI-DC in terms of memory usage, inference speed, FLOPs, and parameter count. All experiments were evaluated on a single-image basis. As shown in Table 7, KFTP requires only marginally higher memory and inference time than TestPromptDC, while maintaining similar FLOPs and the same number of trainable parameters. This indicates that the proposed temporal prompt propagation and Kalman-based update introduce negligible computational overhead. The run times of TestPromptDC and KFTP are close because both perform the same 50-step prompt optimization through the frozen backbone at every frame, which dominates the cost. KFTP adds bidirectional optical-flow estimation, the cycle-consistency check, warping of the prompt and its uncertainty map, and the element-wise update. Together, they add only 4% memory and 1% run time over the frame-wise baseline.

CAPA runs faster per image (0.75 s vs. 5.39 s), but it requires about ten times more memory and optimizes scene-level parameters over a whole sequence, which is less suited to online streaming input. Furthermore, due to compatibility issues with profiling libraries and their dependencies within the experimental environment, the FLOPs and parameter counts for Zero-DC and Marigold-DC could not be measured. Moreover, since both TestPromptDC and our proposed method freeze the backbone model during the test-time adaptation phase and optimize only the lightweight prompt-related parameters, it was observed that both approaches exhibit a comparable number of parameters.

G Comparison with Video Depth Completion Frameworks

We further compare video depth completion frameworks under different supervision and adaptation settings in Table 8. While RayFusion (Kim et al., 2024) and SVDC (Zhu et al., 2025) rely on supervised training with GT, KFTP performs clip-level test-time adaptation without GT. Compared with CAPA (Ke et al., 2026), which conducts scene-level adaptation, KFTP achieves lower average RMSE without requiring costly scene-specific optimization, while CAPA is better on DyDToF (0.707 vs. 0.940).

Table 8: RMSE comparison of video depth completion frameworks under different learning and adaptation settings. † denotes results taken from the original papers (Kim et al., 2024; Zhu et al., 2025). Average‡ indicates the mean RMSE over the six datasets (KITTI, DDAD, vKITTIv2, DyDToF, Sintel, and NYUv2).

Model	Learning Protocol	VOID	Dynamic Replica	Average‡
RayFusion† (Kim et al., 2024)	Supervised Learning w/ GT	0.111	-	-
SVDC† (Zhu et al., 2025)	Supervised Learning w/ GT	-	0.086	-
CAPA (Ke et al., 2026)	Scene-level Adaptation	0.555	**0.110**	2.145
Ours	Clip-level Adaptation	**0.192**	0.120	**1.861**

H Formulation Details

H.1 Derivation of the Prompt Fusion Update

Given $\boldsymbol{\mu}_{t|t-1}$, $\mathbf{z}_t$, and element-wise maps $\boldsymbol{\Sigma}_{t|t-1}, \mathbf{R}_t > 0$, each identified with the diagonal matrix it defines, consider the objective J_t in Equation (23), where $\|\mathbf{x}\|^2_{\mathbf{A}} = \mathbf{x}^\top \mathbf{A}\mathbf{x}$. Its gradient is

$$\nabla J_t(\boldsymbol{\mu}) = \boldsymbol{\Sigma}^{-1}_{t|t-1}(\boldsymbol{\mu} - \boldsymbol{\mu}_{t|t-1}) + \mathbf{R}_t^{-1}(\boldsymbol{\mu} - \mathbf{z}_t). \tag{32}$$

Setting it to zero gives

$$(\boldsymbol{\Sigma}^{-1}_{t|t-1} + \mathbf{R}_t^{-1})\boldsymbol{\mu} = \boldsymbol{\Sigma}^{-1}_{t|t-1}\boldsymbol{\mu}_{t|t-1} + \mathbf{R}_t^{-1}\mathbf{z}_t, \tag{33}$$

whose solution is

$$\boldsymbol{\mu}_t = \boldsymbol{\mu}_{t|t-1} + \mathbf{K}_t \odot (\mathbf{z}_t - \boldsymbol{\mu}_{t|t-1}), \qquad \mathbf{K}_t = \boldsymbol{\Sigma}_{t|t-1} \oslash (\boldsymbol{\Sigma}_{t|t-1} + \mathbf{R}_t), \tag{34}$$

that is, Eqs. (25)–(26). The Hessian $\nabla^2 J_t = \boldsymbol{\Sigma}^{-1}_{t|t-1} + \mathbf{R}_t^{-1}$ is constant and positive definite, so $\boldsymbol{\mu}_t$ is the unique minimizer, and the inverse Hessian is

$$(\nabla^2 J_t)^{-1} = \boldsymbol{\Sigma}_{t|t-1} \odot \mathbf{R}_t \oslash (\boldsymbol{\Sigma}_{t|t-1} + \mathbf{R}_t) = (\mathbf{1} - \mathbf{K}_t) \odot \boldsymbol{\Sigma}_{t|t-1}, \tag{35}$$

that is, Eq. (27). Since J_t is quadratic, it equals its second-order expansion about $\boldsymbol{\mu}_t$,

$$J_t(\boldsymbol{\mu}) = J_t(\boldsymbol{\mu}_t) + \tfrac{1}{2}\|\boldsymbol{\mu} - \boldsymbol{\mu}_t\|^2_{\boldsymbol{\Sigma}_t^{-1}}, \tag{36}$$

so $(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ summarizes J_t up to an additive constant. Because the weights are element-wise, J_t separates into one scalar quadratic per prompt element, and the update is its exact minimizer rather than an approximation. No distributional assumption is used. This equivalence between weighted least squares and the Kalman update is standard (Sorenson, 1970; Humpherys et al., 2012), and the Gaussian model is one route to it rather than a requirement (Uhlmann & Julier, 2024). We claim no novelty for it.

The derivation concerns the per-frame update only. It does not imply that the full prediction–update recursion, including the warping of $\boldsymbol{\Sigma}$ and the addition of $\mathbf{Q}_t$, is the optimum of a single global objective, and it does not assume that $\mathbf{z}_t$ and $\boldsymbol{\mu}_{t|t-1}$ are independent.

H.2 State Propagation, Observation, and Scope

For a fixed flow field, bilinear warping is linear in the warped quantity (Jaderberg et al., 2015), so Eq. (11) is a time-varying linear state propagation that aligns the previous prompt to the current frame and carries it forward. We call $\boldsymbol{\Sigma}_{t|t-1}$ the prior prompt uncertainty map and $\mathbf{R}_t$ the current observation uncertainty map. KFTP propagates only the previous posterior because, under the recursive update, $(\boldsymbol{\mu}_{t-1}, \boldsymbol{\Sigma}_{t-1})$ already summarizes all earlier frames. This keeps memory and per-frame cost constant, as causal online processing requires.

The current-frame prompt observation $\mathbf{z}_t$ is initialized from $\boldsymbol{\mu}_{t|t-1}$ and obtained by test-time optimization through the nonlinear model Φ. It is therefore neither a linear measurement nor independent of the prior, and the fusion assumes neither property. Because $\mathbf{z}_t$ and $\boldsymbol{\mu}_{t|t-1}$ have the same shape, the observation matrix $\mathbf{H}_t$ of Eqs. (5)–(7) is the identity, and the nonlinearity of Φ enters only through Eq. (28). Finally, since $\boldsymbol{\Sigma}_{t|t-1}$ and $\mathbf{R}_t$ are designed uncertainty maps rather than calibrated covariances, we do not claim the optimality of a probabilistic Kalman filter.

I Structure of the Uncertainty Maps

The element-wise uncertainty maps are a design choice for efficiency, not a claim that prompt elements are independent. On ten scenes subsampled from KITTI, Table 9 compares them with local-block variants in which each diagonal entry is replaced by a $k \times k$ block, so that neighboring prompt elements share weighting information. The three variants are within 0.001 RMSE and 0.003 MAE of one another, while the 3×3 and 5×5 blocks increase run time by 10.4% and 32.7%, respectively. Run times differ slightly from Table 7 because they are measured on this subset.

Table 9: Uncertainty structures on ten KITTI scenes. All variants share the frozen backbone, prompt optimization, and evaluation protocol.

Structure	RMSE ↓	MAE ↓	Time (s/img) ↓
Element-wise (Ours)	1.472	0.448	**5.47**
Local block, 3×3	**1.471**	0.449	6.04
Local block, 5×5	**1.471**	**0.446**	7.26

J Temporal Correspondence of Frame-wise Prompts

To test whether prompts carry temporal structure that warping can exploit, we take prompts optimized independently for each frame, without the temporal loss or prior initialization, decompose each into Laplacian pyramid bands (Burt & Adelson, 1983), and measure the cosine similarity between corresponding bands of adjacent frames before and after aligning the previous prompt with optical flow. The similarity is computed over all positions and channels of a band.

Table 10: Band-wise cosine similarity between prompts of adjacent frames, before and after optical-flow alignment. Band 0 has the highest frequency.

Band	KITTI		NYUv2	
	Unaligned	Aligned	Unaligned	Aligned
Band 0	0.059	0.068	0.043	0.034
Band 1	0.059	0.122	0.032	0.090
Band 2	0.090	0.181	0.046	0.134
Band 3	0.157	0.279	0.075	0.187
Residual	**0.344**	**0.462**	**0.246**	**0.306**

Alignment raises the similarity of Bands 1–3 and the residual on both datasets ($+0.063$, $+0.091$, $+0.122$, and $+0.118$ on KITTI, and $+0.058$, $+0.088$, $+0.112$, and $+0.060$ on NYUv2), indicating that part of the frame-to-frame prompt variation is explained by scene motion and can be recovered by warping. The highest-frequency band gains little on KITTI ($+0.009$) and decreases slightly on NYUv2 ($0.043 \rightarrow 0.034$), indicating that not every prompt component propagates reliably. KFTP therefore does not copy the propagated prompt unconditionally: where propagation is unreliable, $\mathbf{Q}_t$ raises $\mathbf{\Sigma}_{t|t-1}$, which raises $\mathbf{K}_t$ and shifts the weight toward the current observation.

K Kalman Filtering for Depth and Dense Prediction

Recursive uncertainty-weighted fusion has a long history in depth estimation. Matthies et al. (1989) maintain a per-pixel depth and variance and refine both with a Kalman filter under known camera motion, Hung & Ho (1999) add local surface structure to a pixel-based Kalman model, and Zhang et al. (2022) integrate a differentiable EKF, which updates IMU-preintegrated ego-motion with visual measurements, into unsupervised depth training. Beyond depth, a deep Kalman filtering network (Lu et al., 2018) restores video frames using the previously restored frame as its prior, and KalmanFlow 2.0 (Bao et al., 2019) tracks optical flow as a time-varying state. These methods filter a task output, so their updates rely on an explicit measurement model, camera motion, or inertial sensing. We do not claim the recursion or the per-pixel gain as novel. KFTP instead filters the visual prompt of a frozen depth foundation model. Since the prior and the observation are prompts of the same shape, KFTP needs no measurement model or camera pose. Its uncertainty is built from cues

available at test time, and every output is a single prediction $\Phi(\mathbf{I}_t + \boldsymbol{\mu}_t)$ of the frozen model rather than a blend of two depth maps. Test-time video depth methods such as CVD (Luo et al., 2020) and RCVD (Kopf et al., 2021) fine-tune a monocular network per video offline with estimated camera poses, whereas KFTP runs online, keeps the network frozen, and uses sparse metric depth.

L LIMITATIONS

The main limitation of KFTP is the cost of test-time optimization. Although the temporal components increase run time by only about 1% over frame-wise prompting, the prompt must still be optimized at every frame, which currently limits its applicability to real-time streaming. Reducing this cost is left for future work. In addition, KFTP propagates prompts with optical flow and does not use camera poses or multi-view geometry. Incorporating them could improve propagation in rigid regions but would require pose estimation and sequence-level processing.