LAGRANGIAN–HAMILTONIAN FLOWS FOR VIDEO PREDICTION AND IMAGE GENERATION: A SYMPLECTIC PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce LHFM, a geometric framework for learning image dynamics that combines Lagrangian image representations with Hamiltonian dynamics. These structures play a central role in classical mechanics, symplectic geometry, and geometric quantization. LHFM represents images as exact Lagrangian graphs and models their evolution through image-dependent Hamiltonian flows. This construction provides a geometric interpretation of image-space dynamics by connecting them to the Hamiltonian evolution of Lagrangian submanifolds. The framework supports both video prediction, our primary application, and image generation. We propose a novel deterministic video prediction method based on the Hamiltonian evolution of Lagrangian image representations. This video variant, LHFM-V, is a recurrent model that achieves the lowest reported FLOP count among the compared recurrent models with similar prediction accuracy. The image variant, LHFM-I, shows that the same construction is compatible with flow matching: in a matched experiment, it attains a lower FID than the flow-matching baseline.

1 INTRODUCTION

Diffusion models and flow-based generative models generate images by transforming a simple distribution into the data distribution (Ho et al., 2020; Song et al., 2021; Lipman et al., 2023). Flow matching (FM) provides a simulation-free training objective for continuous normalizing flows by regressing a time-dependent vector field onto conditional vector fields (Lipman et al., 2023). In the original FM formulation, the network directly models a vector field on the data space where each image represents a point of this space.

A similar perspective applies to video prediction: conditioned on the observed sequence, a learned flow evolves the observed frames into future frames. Thus, image generation and video prediction can both be formulated as learning a flow on image space, although their initial conditions and training objectives differ.

Image dynamics involve both spatial structure (pixel locations) and color (channel values). In standard FM, however, the vector field acts on channel values at fixed pixel locations, so changes in spatial structure are reflected only implicitly through changes in color. This motivates a question: how can these two aspects of image evolution be expressed through a common geometric representation?

We answer this question by introducing *Lagrangian–Hamiltonian Flow Matching* (LHFM), a geometric framework that combines Lagrangian image representations with Hamiltonian dynamics. A Lagrangian submanifold can be viewed as a "generalized point" which encodes the space and phase simultaneously. Motivated by this, we represent each image as an exact Lagrangian graph that jointly encodes pixel locations and channel values. Image evolution is then described by Hamiltonian flows that transport these graphs, as illustrated in Figure 1. Throughout, "Lagrangian" refers to a submanifold rather than to an action functional.

A central issue is compatibility: a Hamiltonian flow preserves the Lagrangian property but need not preserve a graph representation. We show that a Hamiltonian transports a prescribed family of exact

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.

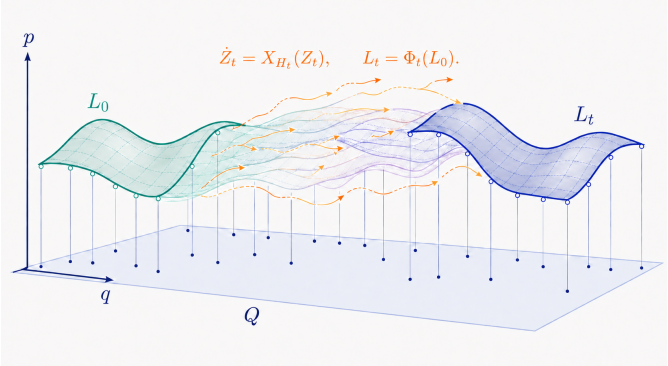

Figure 1: **Hamiltonian deformation of a Lagrangian submanifold.** For a fixed image trajectory, the time-dependent ambient Hamiltonian H_t transports L_0 to $L_t = \Phi_t(L_0)$. The surfaces schematically represent higher-dimensional exact Lagrangians.

Lagrangian graphs if and only if it satisfies a Hamilton–Jacobi condition. This characterization leads to a natural parameterization of the vector field on the data space in terms of a *transport vector field*, which moves spatial locations, and a *source field*, which changes channel values along that motion. The construction is detailed in Section 4.

We instantiate LHFM as LHFM-I for image generation and LHFM-V for deterministic video prediction, where I and V denote image and video, respectively. We evaluate the two variants on unconditional CIFAR-10 generation and Moving MNIST prediction. In a matched comparison, LHFM-I improves upon the standard independent conditional flow-matching baseline. LHFM-V uses a recurrent architecture and has the lowest reported FLOP count among the compared recurrent models with similar prediction accuracy.

The primary focus of this work is deterministic video prediction: LHFM-V is our main method, and the video experiments constitute the main empirical evaluation. LHFM-I is included to show that the same transport–source parameterization is compatible with flow-matching image generation without changing the training objective. We therefore evaluate LHFM-I only in a controlled comparison against a matched I-CFM baseline, and we do not claim state-of-the-art image generation.

Our contributions are:

- to our knowledge, the first representation of images as exact Lagrangian graphs, with a Hamilton–Jacobi characterization (classical in form; proved in Appendix A.3) of the Hamiltonians that transport them;
- a transport–source parameterization derived from it, which contains the standard flow-matching velocity as the case $U_t \equiv 0$; and
- empirical studies on deterministic Moving MNIST prediction, where LHFM-V has the lowest FLOP count among the compared recurrent video predictors (Table 3), and on CIFAR-10 generation, where LHFM-I attains a lower FID than a matched I-CFM baseline, demonstrating compatibility with flow matching.

2 Related Work

Flow matching. Flow matching trains continuous normalizing flows through simulation-free regression of conditional vector fields (Lipman et al., 2023). Conditional flow matching extends this construction to general source and target distributions, with independent and optimal transport couplings as particular choices (Tong et al., 2024). LHFM-I retains the independent conditional flow-matching objective but parameterizes the image velocity through a transport vector field and a source field.

Geometric formulations. Riemannian Flow Matching extends flow matching to manifolds (Chen & Lipman, 2024), while Metric Flow Matching uses data-dependent Riemannian metrics to construct conditional paths (Kapuśniak et al., 2024). LHFM takes a different approach: it represents each image as an exact Lagrangian graph and characterizes Hamiltonian flows that transport these representations. The underlying connection between Lagrangian submanifolds and the Hamilton–Jacobi equation is classical (Cariñena et al., 2006), and transport–source image dynamics also ap-

pear in image metamorphosis (Trouvé & Younes, 2005; Holm et al., 2009). Our contribution applies this connection to image representations and derives a transport–source parameterization of image dynamics.

Video prediction. Recurrent video predictors advance a hidden state frame by frame, from ConvLSTM (Shi et al., 2015) and the PredRNN family (Wang et al., 2017; 2022) to PhyDNet, which constrains part of its latent dynamics with a learned PDE (Le Guen & Thome, 2020). LHFM-V belongs to this category: conditioned on observed frames, it recurrently predicts transport and source fields and evolves the image through their combined dynamics. Recurrent-free predictors (Gao et al., 2022; Tang et al., 2026) and generative video models (Davtyan et al., 2023) are outside the scope of this work.

3 Geometric Preliminaries and Motivation

A symplectic manifold (X^{2m}, ω) is a smooth manifold equipped with a closed, nondegenerate two-form. A submanifold $L \subset X$ is *Lagrangian* if $\dim L = m$ and $\omega|_L = 0$. The cotangent bundle T^*Q carries the canonical one-form λ and symplectic form $\omega = -d\lambda$. For a smooth function $S : Q \to \mathbb{R}$, the graph $\mathrm{graph}(dS)$ is an exact Lagrangian submanifold: the pullback of λ to the graph equals dS (Cannas da Silva, 2001; McDuff & Salamon, 2017).

A time-dependent Hamiltonian $H_t : T^*Q \to \mathbb{R}$ determines a Hamiltonian vector field through $\iota_{X_{H_t}}\omega = d_z H_t$. Its flow preserves the symplectic form and the exactness of Lagrangian submanifolds, but not necessarily their graph representations. Our construction uses exact graphs to represent images and the Hamilton–Jacobi equation to characterize their evolution.

The cotangent bundle T^*Q provides the natural phase space of classical mechanics, with its canonical coordinates representing position and momentum. In semiclassical analysis, an oscillatory state of the form $a(q)e^{\mathrm{i}S(q)/\hbar}$ is microlocally associated with the Lagrangian graph $L_S = \mathrm{graph}(dS)$, on which the phase function determines the relation $p = dS(q)$. Whereas a classical phase-space point specifies position and momentum simultaneously, the Heisenberg uncertainty principle prevents a quantum state from being sharply localized in both variables (Kennard, 1927; Robertson, 1929). This phase-space viewpoint motivates our use of Lagrangian graphs to organize spatial locations, channel values, and their local spatial sensitivities within a common geometric representation.

Table 1: Geometric motivation for LHFM. Hamiltonian evolution refers to the ambient cotangent bundle, not the image-space vector field.

	Classical mechanics	**Semiclassical (WKB)**
Representation	Point $(q, p) \in T^*Q$	Lagrangian graph with amplitude
	Vanilla Flow Matching	**LHFM**
Representation	An image point J	Graph Lagrangian $L_J = \mathrm{graph}(dS_J)$
Dynamics	A general learned Flow in image space	Hamiltonian flow on T^*Q

4 Lagrangian–Hamiltonian Flow Matching

We first define a Lagrangian representation of an image and characterize the Hamiltonian flows that transport these representations. We then use the induced image dynamics for conditional flow matching and deterministic video prediction. The representation and continuous geometric construction are shared; initial conditions, conditioning, losses, and numerical methods are specified separately for the two applications.

4.1 Image Space and Continuous Representation

Let $M = \mathbb{R}^2$ be the spatial coordinate space and $V = \mathbb{R}^n$ the space of n-channel pixel values. For the pixel-centered grid $\mathcal{G}_N$ of an $N \times N$ image, write $\mathcal{X}_N = V^{\mathcal{G}_N}$ and $d = nN^2$. The linear full-

band cosine extension associates each image $I \in \mathcal{X}_N$ with a smooth map $F_I = \mathcal{F}_N(I) : M \to V$ satisfying $F_I|_{\mathcal{G}_N} = I$. The detailed construction and its properties are given in Appendix A.1.

4.2 Lagrangian Image Representation

Introduce an auxiliary dual variable $a \in V^*$. Throughout, we use the standard Euclidean identifications of vector spaces and their duals when writing coordinate transposes. Set $Q = \mathbb{R}^2 \times V^*$, and equip its cotangent bundle T^*Q with canonical coordinates $(x, a; \xi, \eta)$, where ξ and η are dual to x and a, respectively. The canonical one-form and symplectic form are

$$\lambda = \xi^\top dx + \eta^\top da, \qquad \omega = -d\lambda. \tag{1}$$

Using the natural duality pairing between V^* and V, we define the generating function $S_I : Q \to \mathbb{R}$ and the associated LHFM encoder by

$$\begin{aligned} S_I(x, a) &= \langle a, F_I(x) \rangle = a^\top F_I(x), \\ E(I) = L_I &= \operatorname{graph}(dS_I) = \left\{ \big(x, a; DF_I(x)^\top a, F_I(x)\big) : (x, a) \in Q \right\}. \end{aligned} \tag{2}$$

Here $DF_I(x)$ denotes the spatial Jacobian of F_I. As the graph of an exact one-form, L_I is an exact Lagrangian submanifold of T^*Q.

Exactness and injective image recovery are established in Proposition A.4 in Appendix A.2.

Geometric Intuition of Lagrangian representation The auxiliary variable $a \in V^*$ acts as a linear probe of the channel values, with scalar response $S_I(x, a)$ at spatial location x. For an RGB image, $a = e_R$ selects the red channel, whereas $a = (e_G - e_R)/\sqrt{2}$ probes the normalized green-minus-red contrast.

Differentiating with respect to the spatial and probe variables yields

$$dS_I = \underbrace{DF_I(x)^\top a}_{\xi} \cdot dx + \underbrace{F_I(x)}_{\eta} \cdot da.$$

Thus, η records the channel values and determines how the response changes when the probe is varied, while ξ describes how the same response changes under a spatial displacement.

As shown in Figure 2,[1] the covector $\xi(a) = DF_I(x_0)^\top a$, gives the direction of steepest increase of the probe response at x_0. The red arrow ($a = e_R$) indicates the direction in which the red-channel value increases most rapidly, whereas the green arrow ($a = (e_G - e_R)/\sqrt{2}$) indicates the direction of steepest increase in the green-minus-red contrast.

The Lagrangian graph therefore organizes pixel values and their local spatial sensitivities into a single geometric object.

4.3 Hamiltonian Dynamics

We write J_t for an image trajectory, $F_{J_t} = \mathcal{F}_N(J_t)$ for its continuous representation, and $J^{(m)}$ for numerical iterates.

Hamiltonian parameterization. Let $z = (x, a; \xi, \eta) \in T^*Q$ denote the canonical coordinates. For a Hamiltonian, let X_{H_t} be the Hamiltonian vector field defined by $\iota_{X_{H_t}} \omega = d_z H_t$. A Hamiltonian trajectory $z(t) = (x(t), a(t); \xi(t), \eta(t))$ with initial point $z_0 \in T^*Q$ satisfies

$$\frac{d}{dt} z(t) = X_{H_t}\big(z(t)\big), \qquad z(0) = z_0. \tag{3}$$

A Hamiltonian flow preserves the exactness of a Lagrangian submanifold (Viterbo, 2023, Proposition 4.13), but it need not preserve its representation as a graph over the base (Kragh, 2026, Lemma 2.4 and Corollary 2.5). This raises a natural question: which Hamiltonians transport a

[1] Illustration by NoCopyrightGirl (NCG), reproduced and annotated under the artist's published usage rules. The artwork is used solely for illustration and was not used for model training.

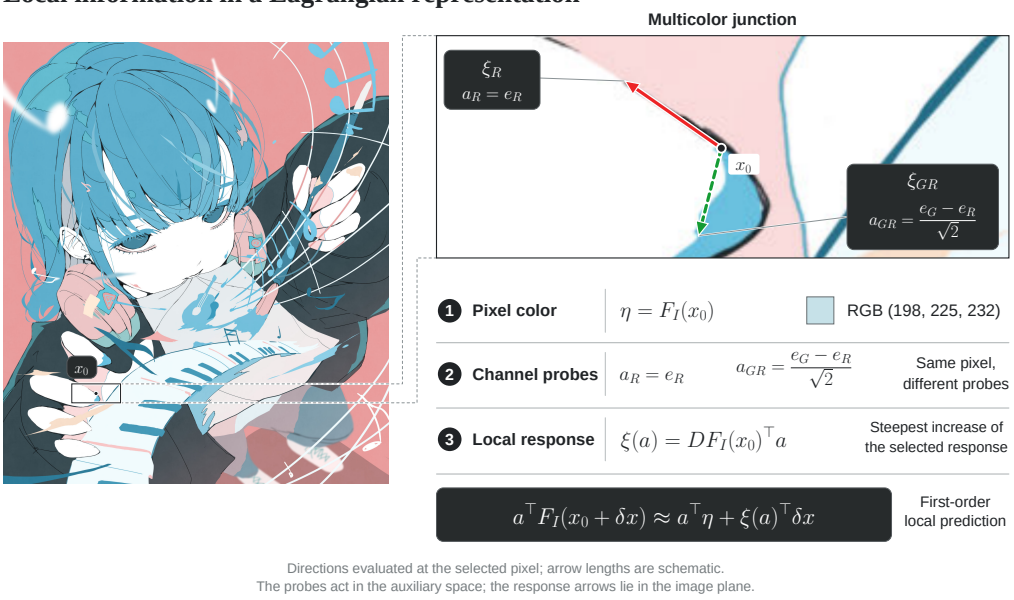

Figure 2: **Local information encoded by the Lagrangian representation.** Arrow directions are computed at x_0 in Euclidean pixel coordinates; lengths are schematic and not to scale.

prescribed family of exact Lagrangian graphs? The following lemma provides a precise characterization: such transport occurs if and only if the generating functions satisfy the associated Hamilton–Jacobi equation up to an additive function of time. This characterization is central to our construction.

Lemma 4.1 (Hamiltonian transport of image graphs)**.** *Let $\mathcal{I} \subset \mathbb{R}$ be an open time interval and let $F \in C^\infty(\mathcal{I} \times \mathbb{R}^2; \mathbb{R}^n)$. Write $F_t(x) = F(t, x)$ and define*

$$S_t(x, a) = a^\top F_t(x), \qquad L_t = \operatorname{graph}(dS_t) = \left\{(x, a; DF_t(x)^\top a, F_t(x)) : (x, a) \in Q\right\}.$$

Let H_t be a smooth time-dependent Hamiltonian on T^*Q. Assume that its Hamiltonian flow $\Phi^H_{s\to t}$ is defined on all of T^*Q for every $s, t \in \mathcal{I}$.

Then the following statements are equivalent:

1. *The Hamiltonian flow transports the prescribed image graphs: $\Phi^H_{s\to t}(L_s) = L_t$ for $s, t \in \mathcal{I}$.*
2. *The generating function S_t and Hamiltonian H_t satisfy the Hamilton–Jacobi equation*
$$\partial_t S_t(q) + H_t(q, d_q S_t(q)) = c(t), \qquad q \in Q, \tag{4}$$
where $c \in C^\infty(\mathcal{I}; \mathbb{R})$ only depends on the time.
3. *There exist smooth maps $U : \mathcal{I} \times T^*Q \to \mathbb{R}^2$ and $B : \mathcal{I} \times T^*Q \to \mathbb{R}^n$ such that*
$$\begin{aligned} H_t(x, a; \xi, \eta) = c(t) - a^\top \partial_t F_t(x) + \big(\xi - DF_t(x)^\top a\big)^\top U_t(x, a; \xi, \eta) \\ + \big(\eta - F_t(x)\big)^\top B_t(x, a; \xi, \eta). \end{aligned} \tag{5}$$

The proof is given in Appendix A.3.

The restricted coefficients U^L_t, B^L_t and the corresponding Hamiltonian characteristics are given in Proposition A.5 in Appendix A.3. For a finite-image trajectory, we apply these results with $F_t = F_{J_t}$ to derive the induced image velocity from the equation for η.

Induced image dynamics. Let $J_t \in \mathcal{X}_N$ be a smooth image path and set $F_{J_t} = \mathcal{F}_N(J_t)$. Suppose that the Hamiltonian flow transports the graphs L_{J_t}, as characterized by Lemma 4.1. By Hamilton's equation $\dot\eta = -\partial_a H_t$ (equation 19), we define

$$R^L_t(x, a) := -\partial_a H_t\big(x, a; DF_{J_t}(x)^\top a, F_{J_t}(x)\big).$$

Along a Hamiltonian trajectory, $\eta_t = F_{J_t}(x_t)$ and $\dot{x}_t = U_t^L(x_t, a_t)$. Proposition A.5 therefore gives

$$R_t^L(x_t, a_t) = \dot{\eta}_t = \frac{d}{dt} F_{J_t}(x_t) = \partial_t F_{J_t}(x_t) + DF_{J_t}(x_t) U_t^L(x_t, a_t). \tag{6}$$

Equivalently, for every $(x, a) \in Q$,

$$\partial_t F_{J_t}(x) = R_t^L(x, a) - DF_{J_t}(x) U_t^L(x, a).$$

Although the two terms may depend on a, their difference is independent of a by equation 6. For the parameterization used below, the probe remains constant and spatial transport is independent of the probe:

$$U_t^L(x, a) = U_t(x), \qquad B_t^L(x, a) = 0.$$

It follows that R_t^L is also independent of a; write it as $R_t(x)$. The induced continuous image velocity is

$$V_t := \partial_t F_{J_t} = R_t - dF_{J_t}(U_t) = R_t - DF_{J_t} U_t. \tag{7}$$

Equation 7 is a vector-valued transport equation with a source term, governing the continuous image representation. Accordingly, we refer to $U_t \in \Gamma(TM)$ as the *transport vector field* and $R_t \in C^\infty(M; V)$ as the *source field*, while DF_{J_t} denotes the spatial Jacobian. As expressed by equation 6, the channel values change at rate R_t along trajectories satisfying $\dot{x}_t = U_t(x_t)$.

Restricting to the pixel grid yields

$$u_t := U_t|_{\mathcal{G}_N}, \qquad r_t := R_t|_{\mathcal{G}_N}, \qquad v_t := \dot{J}_t = r_t - \big(DF_{J_t}|_{\mathcal{G}_N}\big) u_t. \tag{8}$$

All Jacobian–vector products on the grid are pointwise. Lower-case u_t, r_t, v_t denote grid arrays, whereas upper-case U_t, R_t, V_t denote continuous fields.

Scope. Every C^1 image path admits a Hamiltonian realization for any transport field (Proposition A.6); standard flow matching is the case $U_t \equiv 0$, and the split is not unique (Remark A.8). The geometry thus does not restrict image velocities; it derives the transport–source parameterization, with the probe a packaging all channels into one generating function $S_I = a^\top F_I$ governed by one Hamiltonian.

5 Models

From equation 7, Hamiltonian dynamics naturally induce a decomposition into spatial transport and changes in channel values. The transport vector field moves image structures, while the source field changes their appearance along the motion.

Figure 3 illustrates these complementary roles: the chosen source field changes the digit's color, while transport moves its spatial structure.

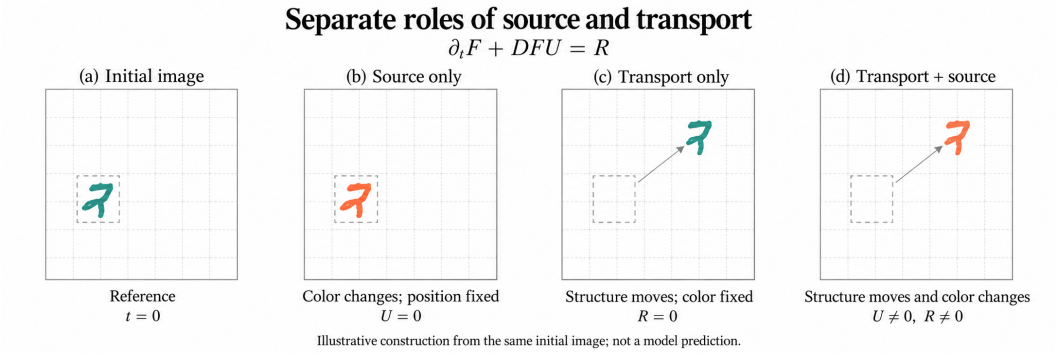

Figure 3: Separate roles of source and transport.

Before introducing the models, we illustrate the benefits of such decomposition with a simple example in Figure 4. Pixelwise linear interpolation blends features at fixed spatial locations, producing

overlapping facial features when the endpoints are not aligned. In contrast, the transport–source construction aligns corresponding features while changing their appearance. This comparison illustrates how the Hamiltonian dynamics can capture structural changes through explicit spatial transport, motivating its use in image generation and video prediction.

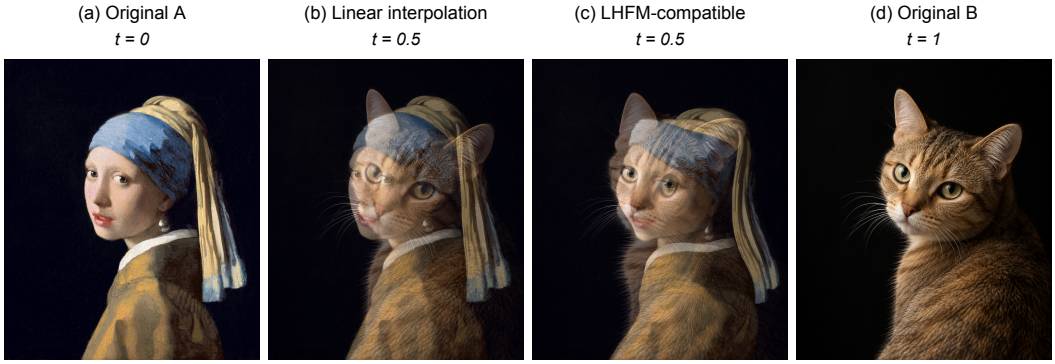

Figure 4: Spatial transport versus pixelwise blending. (a,d) Endpoint images: *Girl with a Pearl Earring* and an AI-generated cat portrait. (b) Pixelwise linear interpolation at $t = 0.5$. (c) Interpolation compatible with the LHFM transport–source formulation at the same time.

5.1 LHFM-I: Flow Matching for Image Generation

Let ν_{data} be the image data distribution and $\nu_0 = \mathcal{N}(0, \text{Id}_d)$ the prior. For independent $\epsilon \sim \nu_0$, $I \sim \nu_{\text{data}}$, and $t \sim \mathcal{U}(0,1)$, define the conditional interpolation

$$J_t^{\text{cond}} = (1-t)\epsilon + tI, \qquad y = I - \epsilon. \tag{9}$$

A neural network with parameters θ parameterizes a source array $r_\theta(t, J)$ and a continuous transport vector field $U_\theta(t, J, \cdot)$. Its grid restriction is $u_\theta(t, J) := U_\theta(t, J, \cdot)|_{\mathcal{G}_N}$. The induced image velocity is

$$v_\theta(t, J) = r_\theta(t, J) - \big(DF_J|_{\mathcal{G}_N}\big)u_\theta(t, J).$$

We minimize the conditional flow-matching objective

$$\mathcal{L}_{\text{CFM}}(\theta) = \frac{1}{d}\mathbb{E}\left\|v_\theta(t, J_t^{\text{cond}}) - y\right\|_2^2. \tag{10}$$

This is the standard independent-endpoint regression objective (Lipman et al., 2023; Tong et al., 2024), with a transport–source parameterization of the predicted image velocity.

For sampling, draw $J_0 \sim \nu_0$ and solve $\dot{J}_t = v_{\bar{\theta}}(t, J_t)$, $0 \le t \le 1$, using exponential moving average (EMA) parameters $\bar{\theta}$. The implementation evaluates the full-band DCT spatial Jacobian and uses a fixed-step Heun solver in image coordinates.

Algorithm 1 LHFM-I for Image Generation

Input: Prior ν_0, data distribution ν_{data}, and parameterized transport and source arrays u_θ, r_θ.
Velocity:
$v_\theta(t, J) := r_\theta(t, J) - \big(DF_J|_{\mathcal{G}_N}\big)u_\theta(t, J)$,
where $F_J = \mathcal{F}_N(J)$.
while training **do**
  Sample independently $\epsilon \sim \nu_0$, $I \sim \nu_{\text{data}}$, $t \sim \mathcal{U}(0,1)$
  $J \leftarrow (1-t)\epsilon + tI$
  $\mathcal{L}(\theta) \leftarrow d^{-1}\|v_\theta(t, J) - (I - \epsilon)\|_2^2$
  $\theta \leftarrow \text{Update}(\theta, \nabla_\theta \mathcal{L}(\theta))$
return v_θ

5.2 LHFM-V: Deterministic Video Prediction

Let $\mathbf{I}_{\text{obs}} = (I_1, \ldots, I_{T_{\text{obs}}})$ be the observed frames. We predict the next K frames by evolving an image state from $J_{T_{\text{obs}}} = I_{T_{\text{obs}}}$. Here s denotes video time, measured in frame intervals, rather than the noise-to-data time t in conditional flow matching. The continuous formulation is

$$\partial_s F_{J_s} = R_s - DF_{J_s}U_s, \qquad J_{T_{\text{obs}}} = I_{T_{\text{obs}}}, \tag{11}$$

with fields conditioned on $\mathbf{I}_{\text{obs}}$ and the evolving image and recurrent states.

Conditional prediction and numerical integration. LHFM-V encodes the observed frames and their temporal differences with a convolutional history encoder. It jointly predicts features for the future frames, then recurrently updates an image state using transport and source outputs. A causal memory stores the observed history and recent predicted states; feedback from the current image updates the recurrent state at each half-frame interval. The field prediction modules share parameters across the rollout. Unlike the bounded transport used in LHFM-I, the LHFM-V transport output has no explicit amplitude bound or temporal gate. The Moving MNIST implementation uses $T_{\mathrm{obs}} = K = 10$ and two steps per future frame. Set $h = 1/2$, $s_m = T_{\mathrm{obs}} + mh$, and $J^{(0)} = I_{T_{\mathrm{obs}}}$.

For video prediction we use a second-order upwind spatial discretization, not the DCT derivative used for image generation. The network outputs w_m, in pixels per frame, and r_m^{net}, a source array. The corresponding normalized-coordinate transport is $u_m = w_m/N$. Let $A(w_m)$ denote the upwind approximation to $-U_{s_m} \cdot \nabla$, with zero exterior values. After spatial discretization of equation 11, we hold the predicted transport and source fields fixed over each interval of length h. Solving the resulting linear ODE by the variation-of-constants formula gives

$$J^{(m+1)} \approx e^{hA(w_m)} J^{(m)} + \int_0^h e^{(h-\sigma)A(w_m)} r_m^{\mathrm{net}} \, d\sigma, \qquad \widehat{I}_{T_{\mathrm{obs}}+k} = J^{(2k)}. \tag{12}$$

The fields are recomputed from the updated recurrent state at every half-step; the source is integrated along with transport, not added only at the end of the interval. Here r^{net} distinguishes the numerical source output from $r = R|_{\mathcal{G}_N}$ in the exact DCT-based equation 8. Their relation is given in Appendix A.4.

6 Experiments

We evaluate LHFM-I on unconditional CIFAR-10 generation and LHFM-V on deterministic Moving MNIST prediction. Architectures, training settings, and evaluation protocols are provided in Appendix B.

6.1 Image Generation

Table 2 compares LHFM-I with standard independent conditional flow matching (I-CFM) (Tong et al., 2024) and published baselines. At 150k updates, LHFM-I achieves an FID of 3.5809, compared with 3.8241 for the matched I-CFM baseline, a 6.36% reduction in this comparison. The gap persists across sampling budgets: at 64, 128, and 256 NFE, LHFM-I lowers FID by 0.28, 0.28, and 0.24, respectively (Figure 5). Published results are included for context rather than as matched-protocol comparisons.

Table 2: Unconditional CIFAR-10 generation. Published results are taken from Table 5 of Tong et al. (2024). The bottom two rows report our matched 150k-update experiments; boldface highlights LHFM-I.

Note. Published results use different training and evaluation protocols; details are given in Appendix B.4. A dash denotes an unreported value.

Method	**FID ↓**	**NFE**
DDPM (reported)	7.48	274
OT-FM (reported)	6.35	142
VP-FM (reported)	8.06	183
S.I. (reported)	10.27	—
OT-FM (reproduced)	11.527	139.83
VP-FM (Tong et al.)	4.335	525.92
OT-FM (Tong et al.)	3.655	143.00
S.I. (Tong et al.)	4.009	146.12
I-CFM (Tong et al.)	3.659	146.42
OT-CFM (Tong et al.)	3.577	133.94
I-CFM (matched baseline)	3.8241	256
LHFM-I (ours)	**3.5809**	**256**

6.2 Video Prediction

We consider Moving MNIST (Srivastava et al., 2015) prediction from ten observed to ten future frames. Table 3 compares LHFM-V with published recurrent video predictors in terms of model

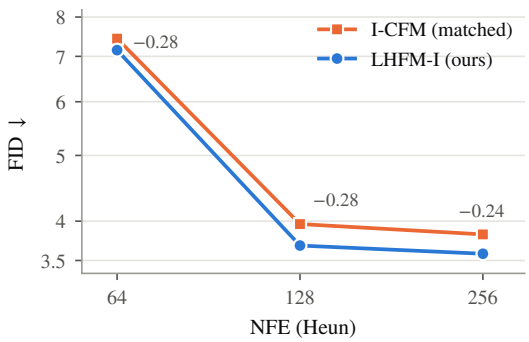

Figure 5: FID versus sampling budget for the matched 150k-update models on unconditional CIFAR-10 (fixed-step Heun solver). Labels give the FID reduction of LHFM-I relative to I-CFM at each budget.

size, computational cost, and prediction accuracy. These results provide context rather than protocol-matched comparisons.

Table 3: Moving MNIST results for recurrent video predictors, which capture temporal dependencies through recurrent hidden-state updates. Baseline numbers are as reported in the original papers.

Method	**Params** (M)	**FLOPs** (G)	**MSE** ↓	**MAE** ↓	**SSIM** ↑
ConvLSTM (Shi et al., 2015)	15.0	56.8	103.3	182.9	0.707
PredRNN (Wang et al., 2017)	23.8	116.0	56.8	126.1	0.867
PredRNN++ (Wang et al., 2018)	38.6	171.7	46.5	106.8	0.898
MIM (Wang et al., 2019b)	38.0	179.2	44.2	101.1	0.910
E3D-LSTM (Wang et al., 2019a)	51.0	298.9	41.3	86.4	0.910
PhyDNet (Le Guen & Thome, 2020)	3.1	15.3	24.4	70.3	0.947
MAU (Chang et al., 2021)	4.5	17.8	27.6	86.5	0.937
PredRNNv2 (Wang et al., 2022)	24.6	708.0	48.4	129.8	0.891
SwinLSTM (Tang et al., 2023)	20.2	69.9	17.7	—	0.962
LHFM-V (ours)	18.6	**13.1**	18.6	62.5	0.958

Why recurrent predictors. LHFM-V shares the defining structure of recurrent predictors: a shared cell advances an explicit state, here the image itself, frame by frame, so each future frame depends only on the observed frames and previously predicted states. We therefore compare within this category.

A discussion of computational cost. Among the recurrent models in Table 3, LHFM-V has the lowest reported FLOP count (13.1G FLOPs). This corresponds to its image-update mechanism: instead of repeatedly reconstructing frames from latent features, LHFM-V predicts transport and source fields that describe motion and color changes. A numerical solver then evolves the current image under these fields, replacing high-resolution neural image decoding with lightweight field prediction and structured image updates.

7 Conclusion

We introduce a geometric framework to image generation and video prediction that represents images as exact Lagrangian graphs and uses their Hamiltonian evolution to formulate transport–source dynamics. This construction connects a geometric representation of images to a recurrent prediction model: learned transport and source fields jointly advance the image through numerical integration. Among the compared recurrent predictors, LHFM-V attains the lowest FLOP count at similar prediction accuracy. The same geometric formulation also supports flow matching for image generation, where our matched experiment shows improved performance over the baseline.

The current model has not yet undergone systematic optimization of its architecture or training strategy. We expect further refinement of these components to improve the accuracy–efficiency trade-off, a direction that remains to be validated in future work.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, mathematical checks, software development, and feedback on experimental design. An AI-generated portrait was used in an illustrative figure, not as training or evaluation data. The author reviewed the AI-assisted material and takes responsibility for the final content and claims.

ETHICS STATEMENT

This work studies image generation and video prediction using public image and video benchmarks and does not involve human participants or private data. As with other image generators, downstream deployment may reproduce dataset biases or enable misleading synthetic content; such uses require application-specific evaluation and safeguards.

REPRODUCIBILITY STATEMENT

Section 4 specifies the image representation, Hamiltonian dynamics, and separate objectives for image generation and video prediction. Appendix A contains the proofs and the relation between continuous geometry and numerical discretization. Appendix B records model architectures, training settings, data splits, checkpoint selection, and metric definitions.

The supplementary materials include the implementation, run-specific configurations, checkpoint identifiers, and evaluation reports for the reported experiments. They also provide the numerical integration and metric computation routines needed to reproduce the evaluations.

REFERENCES

N. Ahmed, T. Natarajan, and K. R. Rao. Discrete cosine transform. *IEEE Transactions on Computers*, C-23(1):90–93, 1974. doi: 10.1109/T-C.1974.223784.

Ana Cannas da Silva. *Lectures on Symplectic Geometry*, volume 1764 of *Lecture Notes in Mathematics*. Springer, 2001. doi: 10.1007/978-3-540-45330-7.

José F. Cariñena, Xavier Gràcia, Giuseppe Marmo, Eduardo Martínez, Miguel C. Muñoz-Lecanda, and Narciso Román-Roy. Geometric Hamilton–Jacobi theory. *International Journal of Geometric Methods in Modern Physics*, 3:1417–1458, 2006. doi: 10.1142/S0219887806001764.

Zheng Chang, Xinfeng Zhang, Shanshe Wang, Siwei Ma, Yan Ye, Xinguang Xiang, and Wen Gao. MAU: A motion-aware unit for video prediction and beyond. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/e25cfa90f04351958216f97e3efdabe9-Abstract.html.

Ricky T. Q. Chen and Yaron Lipman. Flow matching on general geometries. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2302.03660.

Aram Davtyan, Sepehr Sameni, and Paolo Favaro. Efficient video prediction via sparsely conditioned flow matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2211.14575.

Zhangyang Gao, Cheng Tan, Lirong Wu, and Stan Z. Li. SimVP: Simpler yet better video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. URL https://arxiv.org/abs/2206.05099.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, 2020.

Darryl D. Holm, Alain Trouvé, and Laurent Younes. The Euler–Poincaré theory of metamorphosis. *Quarterly of Applied Mathematics*, 67(4):661–685, 2009. doi: 10.1090/S0033-569X-09-01134-2.

Kacper Kapuśniak, Peter Potaptchik, Teodora Reu, Leo Zhang, Alexander Tong, Michael Bronstein, Avishek Joey Bose, and Francesco Di Giovanni. Metric flow matching for smooth interpolations on the data manifold. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/file/f381114cf5aba4e45552869863deaaa7-Paper-Conference.pdf.

Earle H. Kennard. Zur quantenmechanik einfacher bewegungstypen. *Zeitschrift für Physik*, 44(4–5): 326–352, 1927. doi: 10.1007/BF01391200.

Thomas Kragh. Generating functions in $\mathbb{R}^{2n}$ and the Hatcher–Waldhausen map. *Inventiones mathematicae*, 2026. doi: 10.1007/s00222-026-01428-2. URL https://doi.org/10.1007/s00222-026-01428-2.

Vincent Le Guen and Nicolas Thome. Disentangling physical dynamics from unknown factors for unsupervised video prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11474–11484, 2020. URL https://openaccess.thecvf.com/content_CVPR_2020/html/Le_Guen_Disentangling_Physical_Dynamics_From_Unknown_Factors_for_Unsupervised_Video_Prediction_CVPR_2020_paper.html.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

Dusa McDuff and Dietmar Salamon. *Introduction to Symplectic Topology*. Oxford University Press, 3 edition, 2017.

Howard P. Robertson. The uncertainty principle. *Physical Review*, 34(1):163–164, 1929. doi: 10.1103/PhysRev.34.163.

Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, volume 28, 2015. URL https://arxiv.org/abs/1506.04214.

Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Nitish Srivastava, Elman Mansimov, and Ruslan Salakhutdinov. Unsupervised learning of video representations using LSTMs. In *Proceedings of the 32nd International Conference on Machine Learning*, 2015. URL https://proceedings.mlr.press/v37/srivastava15.html.

Gilbert Strang. The discrete cosine transform. *SIAM Review*, 41(1):135–147, 1999. doi: 10.1137/S0036144598336745.

Song Tang, Chuang Li, Pu Zhang, and RongNian Tang. SwinLSTM: Improving spatiotemporal prediction accuracy using swin transformer and LSTM. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13470–13479, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Tang_SwinLSTM_Improving_Spatiotemporal_Prediction_Accuracy_using_Swin_Transformer_and_LSTM_ICCV_2023_paper.html.

Yujin Tang, Lu Qi, Xiangtai Li, Chao Ma, and Ming-Hsuan Yang. Video prediction transformers without recurrence or convolution. *Transactions on Machine Learning Research*, 2026. URL https://arxiv.org/abs/2410.04733.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=CD9Snc73AW.

Alain Trouvé and Laurent Younes. Metamorphoses through Lie group action. *Foundations of Computational Mathematics*, 5(2):173–198, 2005. doi: 10.1007/s10208-004-0128-z.

Claude Viterbo. Symplectic topology in the cotangent bundle through generating functions, 2023. URL https://www.imo.universite-paris-saclay.fr/~viterbo/Cours-M2-2021/Quanti-chapters.pdf. Lecture notes, Paris, Spring 2021; version of February 13, 2023.

Yunbo Wang, Mingsheng Long, Jianmin Wang, Zhifeng Gao, and Philip S. Yu. PredRNN: Recurrent neural networks for predictive learning using spatiotemporal LSTMs. In *Advances in Neural Information Processing Systems*, volume 30, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/e5f6ad6ce374177eef023bf5d0c018b6-Abstract.html.

Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S. Yu. PredRNN++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 5123–5132, 2018. URL https://proceedings.mlr.press/v80/wang18b.html.

Yunbo Wang, Lu Jiang, Ming-Hsuan Yang, Li-Jia Li, Mingsheng Long, and Li Fei-Fei. Eidetic 3D LSTM: A model for video prediction and beyond. In *International Conference on Learning Representations*, 2019a. URL https://openreview.net/forum?id=B1lKS2AqtX.

Yunbo Wang, Jianjin Zhang, Hongyu Zhu, Mingsheng Long, Jianmin Wang, and Philip S. Yu. Memory in memory: A predictive neural network for learning higher-order non-stationarity from spatiotemporal dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9154–9162, 2019b. URL https://openaccess.thecvf.com/content_CVPR_2019/html/Wang_Memory_in_Memory_A_Predictive_Neural_Network_for_Learning_Higher-Order_CVPR_2019_paper.html.

Yunbo Wang, Haixu Wu, Jianjin Zhang, Zhifeng Gao, Jianmin Wang, Philip S. Yu, and Mingsheng Long. PredRNN: A recurrent neural network for spatiotemporal predictive learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. doi: 10.1109/TPAMI.2022.3165153. URL https://arxiv.org/abs/2103.09504. Early access.

APPENDIX CONTENTS

A Mathematical Background and Proofs 14
A.1 Full-band interpolation and exact recovery 14
A.2 Exact Lagrangian Image Graphs 15
A.3 Hamiltonians transporting image graphs 15
A.4 Compatible Continuous and Discrete Image Dynamics 16

B Supplementary Experiments 18
B.1 Image-Generation Architectures and Hyperparameters 18
B.2 Video-Prediction Architectures and Hyperparameters 18
B.3 Numerical Implementation 20
B.4 Evaluation Protocols 21

C Examples 22
C.1 LHFM-I: Image Generation 22
C.2 LHFM-V: Video Prediction 23

A Mathematical Background and Proofs

This appendix contains the results needed for the image representation and Hamiltonian dynamics in Section 4. Interpolation and exact recovery are established first, followed by the graph-transport proofs and the relation between continuous and discrete image dynamics.

A.1 Full-band interpolation and exact recovery

Let $M = \mathbb{R}^2$ be the spatial coordinate space and $V = \mathbb{R}^n$ the space of n-channel pixel values. For images of resolution $N \times N$, the pixel-centered grid and image state space are

$$\mathcal{G}_N = \left\{ \left(\frac{j+1/2}{N}, \frac{\ell+1/2}{N} \right) : j, \ell = 0, \dots, N-1 \right\}, \quad \mathcal{X}_N = V^{\mathcal{G}_N} \cong \mathbb{R}^{nN^2}. \tag{13}$$

Thus an image $I \in \mathcal{X}_N$ assigns a channel vector to each pixel center. We write $d = nN^2$; our CIFAR-10 implementation uses $n = 3$ and $N = 32$, while Moving MNIST uses $n = 1$ and $N = 64$.

The following theorem records the standard full-band interpolation construction based on the orthonormal type-II discrete cosine transform (DCT-II) (Ahmed et al., 1974; Strang, 1999). We record it here to fix the conventions, the proof is standard and thus omitted.

Theorem A.1 (Exact DCT interpolation)**.** *Let $n, N \geq 1$ be integers and let $I \in \mathbb{R}^{n \times N \times N}$. On the pixel-centered grid $x_j = (j + \frac{1}{2})/N$, $j = 0, \dots, N-1$, define*

$$b_0(x) = N^{-1/2}, \qquad b_k(x) = \sqrt{\frac{2}{N}} \cos(\pi k x), \quad k = 1, \dots, N-1. \tag{14}$$

Set $B = (b_k(x_j))_{j,k=0}^{N-1} \in \mathbb{R}^{N \times N}$. For each channel c, let I_c be its pixel array, set $C_c = B^\top I_c B$, and define, for $x = (x_1, x_2) \in \mathbb{R}^2$,

$$F_I^c(x) = \sum_{k,\ell=0}^{N-1} (C_c)_{k\ell}\, b_k(x_1) b_\ell(x_2), \qquad c = 1, \dots, n. \tag{15}$$

Then B is orthogonal, and the map $F_I = (F_I^1, \dots, F_I^n)^\top : \mathbb{R}^2 \to \mathbb{R}^n$ is smooth, even, and 2-periodic in each spatial coordinate. It interpolates the image exactly:

$$F_I^c(x_j, x_\ell) = (I_c)_{j\ell}, \qquad c = 1, \dots, n, \quad j, \ell = 0, \dots, N-1. \tag{16}$$

In particular, $I \mapsto F_I$ is linear and injective, with all DCT coefficients retained, including the constant component.

The even, 2-periodic boundary convention is part of the chosen representation, rather than an assumption about the unknown underlying continuum data distribution.

Definition A.2 (Image function space)**.** For $m \geq 1$, let

$$\mathcal{H}_N^{(m)} = \operatorname{span}\{ b_k(x_1) b_\ell(x_2) e_c : 0 \leq k, \ell < N,\ 1 \leq c \leq m \} \subset C^\infty(M; \mathbb{R}^m),$$

with the subspace topology inherited from the usual Fréchet topology. For $m = n$, write $\mathcal{H}_N = \mathcal{H}_N^{(n)}$. The corresponding grid space is $\mathcal{X}_N^{(m)} = (\mathbb{R}^m)^{\mathcal{G}_N}$.

Proposition A.3 (Extension, recovery, and differentiation)**.** *The componentwise extension $\mathcal{F}_N^{(m)} : \mathcal{X}_N^{(m)} \to \mathcal{H}_N^{(m)}$ is a topological linear isomorphism. In particular, for $m = n$,*

$$\mathcal{F}_N : \mathcal{X}_N \xrightarrow{\cong} \mathcal{H}_N, \qquad \mathcal{F}_N(I) = F_I, \tag{17}$$

with inverse grid sampling $\mathcal{S}_N(F) = F|_{\mathcal{G}_N}$. For a C^1 image path J_t,

$$\partial_t F_{J_t} = D\mathcal{F}_N(J_t)[\dot{J}_t] = \mathcal{F}_N(\dot{J}_t). \tag{18}$$

Proof. Theorem A.1 proves injectivity and exact recovery. The tensor-product basis spans $\mathcal{H}_N^{(m)}$, so grid sampling also gives a right inverse. The extension is continuous because it is a finite expansion in fixed smooth functions, and point evaluation is continuous in the Fréchet topology. Linearity implies $D\mathcal{F}_N(J)[h] = \mathcal{F}_N(h)$, and the chain rule gives the path identity. All these statements concern a fixed resolution N. □

The spatial differential is a different map: $dF_J|_x : T_xM \to V$ acts on spatial tangent vectors, whereas $D\mathcal{F}_N(J)$ acts on image-array perturbations. For a spatial path x_t, the ordinary chain rule gives $\frac{d}{dt}F_{J_t}(x_t) = \mathcal{F}_N(\dot{J}_t)(x_t) + DF_{J_t}(x_t)\dot{x}_t$.

A.2 Exact Lagrangian Image Graphs

Recall that an embedding $\iota : L \hookrightarrow (T^*Q, \omega = -d\lambda)$ is Lagrangian if $\dim L = \dim Q$ and $\iota^*\omega = 0$. It is exact if $\iota^*\lambda = df$ for a globally defined function $f : L \to \mathbb{R}$. In canonical coordinates, $\lambda = p^\top dq$ and $\omega = \sum_i dq^i \wedge dp_i$.

Our convention $\iota_{X_{H_t}}\omega = d_z H_t$ gives

$$\dot{q} = \partial_p H_t, \qquad \dot{p} = -\partial_q H_t, \qquad q = (x, a), \quad p = (\xi, \eta). \tag{19}$$

These are ambient derivatives, taken before restricting to a graph.

Proposition A.4 (Exactness and image recovery)**.** *For every $I \in \mathcal{X}_N$, the graph $L_I = \mathrm{graph}(dS_I)$ is an exact Lagrangian submanifold of T^*Q. The encoder E is injective, and I is recovered by reading η at $a = 0$ and $x \in \mathcal{G}_N$.*

Proof. For $\iota_I(x, a) = (x, a; DF_I(x)^\top a, F_I(x))$,

$$\iota_I^*\lambda = a^\top DF_I(x)\, dx + F_I(x)^\top da = d\big(a^\top F_I(x)\big) = dS_I.$$

The graph is diffeomorphic to Q and has half the dimension of T^*Q. At $a = 0$ and $x \in \mathcal{G}_N$, one has $\xi = 0$ and $\eta = F_I(x) = I(x)$ by Theorem A.1. Thus two coincident graphs have the same image samples, proving injectivity. □

Constant channel offsets are retained through η. Although an individual graph has dimension $n + 2$, its image family has nN^2 degrees of freedom; this representation is not a compression to $n + 2$ scalar coordinates. In image generation, the prior is a distribution over whole image arrays and hence whole encoded graphs, not a distribution of individual points in T^*Q.

A.3 Hamiltonians transporting image graphs

Proof of Lemma 4.1. Write $q = (x, a)$ and $p = (\xi, \eta)$. The moving graph is specified by

$$p = d_q S_t(q).$$

Using Hamilton's equations

$$\dot{q} = \partial_p H_t, \qquad \dot{p} = -\partial_q H_t,$$

the extended vector field $\partial_t + X_{H_t}$ is tangent to this moving graph if and only if

$$-\partial_q H_t(q, d_q S_t) = \partial_t d_q S_t + D_q^2 S_t\, \partial_p H_t(q, d_q S_t).$$

By the chain rule, this is equivalent to

$$d_q[\partial_t S_t + H_t(q, d_q S_t)] = 0.$$

Since $Q = \mathbb{R}^2 \times \mathbb{R}^n$ is connected, the expression in brackets is a function of time alone. This proves the equivalence between tangency and equation 4.

Under the assumed existence of the Hamiltonian flow, tangency and uniqueness of trajectories imply

$$\Phi^H_{s\to t}(L_s) \subseteq L_t.$$

Applying the inverse flow $\Phi^H_{t\to s}$ gives the reverse inclusion, proving statement (1) of Lemma 4.1. Conversely, differentiating this graph-transport identity gives tangency. Thus statements (1) and (2) are equivalent.

To prove that statement (2) implies statement (3), set

$$p_t(q) = d_q S_t(q), \qquad K_t(q, p) = H_t(q, p) - c(t) + \partial_t S_t(q).$$

Equation 4 gives

$$K_t(q, p_t(q)) = 0.$$

The fundamental theorem of calculus along the momentum fiber therefore yields

$$\begin{aligned} K_t(q,p) &= \int_0^1 \partial_p K_t\big(q, p_t(q) + s[p - p_t(q)]\big)^\top [p - p_t(q)]\, ds \\ &= [p - p_t(q)]^\top Z_t(q,p), \end{aligned}$$

where

$$Z_t(q,p) = \int_0^1 \partial_p K_t\big(q, p_t(q) + s[p - p_t(q)]\big)\, ds$$

is smooth. Denote the ξ- and η-components of Z_t by U_t and B_t, respectively. Using

$$p_t(q) = \big(DF_t(x)^\top a, F_t(x)\big)$$

gives equation 5. Conversely, restricting equation 5 to L_t immediately gives equation 4. □

Proposition A.5 (Hamiltonian characteristics on image graphs)**.** *Under the assumptions of Lemma 4.1, suppose that its equivalent conditions hold, and let U_t and B_t be the coefficients in equation 5. Define*

$$U_t^L(x,a) = U_t\big(x, a; DF_t(x)^\top a, F_t(x)\big), \quad B_t^L(x,a) = B_t\big(x, a; DF_t(x)^\top a, F_t(x)\big).$$

The Hamiltonian characteristics restricted to L_t are

$$\begin{aligned} &\dot{x} = U_t^L(x,a), \qquad \dot{a} = B_t^L(x,a), \\ &\dot{\eta} = \partial_t F_t(x) + DF_t(x) U_t^L(x,a), \\ &\dot{\xi} = \big(D\partial_t F_t(x)\big)^\top a + D_x\big(DF_t(x)^\top a\big) U_t^L(x,a) + DF_t(x)^\top B_t^L(x,a). \end{aligned} \tag{20}$$

Here D denotes differentiation with respect to x, and $D_x(DF_t(x)^\top a)$ is computed with a held fixed.

Proof of Proposition A.5. Differentiate equation 5 in the ambient phase space and then restrict to L_t. All terms multiplied by $\xi - DF_t(x)^\top a$ or $\eta - F_t(x)$ vanish, yielding equation 20. In particular,

$$\dot{\eta} = \frac{d}{dt} F_t(x(t)), \qquad \dot{\xi} = \frac{d}{dt}\big[DF_t(x(t))^\top a(t)\big],$$

which explicitly verifies preservation of the graph constraints. □

The function $c(t)$ in the Hamilton–Jacobi condition expresses the freedom to add a function of time to a generating function: replacing S_t by $S_t - \int_{t_0}^t c(\sigma)\, d\sigma$ leaves its graph unchanged and sets the right-hand side to zero. Graph transport permits reparameterization along the submanifold; points with fixed base coordinate need not be Hamiltonian trajectories (Cariñena et al., 2006).

A.4 Compatible Continuous and Discrete Image Dynamics

Proposition A.6 (Compatible Hamiltonian realization)**.** *Let J_t be a C^1 image path with $v_t = \dot{J}_t$. Let U_t be continuous in time, smooth in space, with jointly continuous spatial derivatives, and assume its spatial trajectories exist throughout the interval for every initial time and point. Define*

$$R_t = \mathcal{F}_N(v_t) + DF_{J_t} U_t, \qquad H_t = \xi^\top U_t - a^\top R_t. \tag{21}$$

Then the Hamiltonian flow transports $L_{J_{t_0}}$ to L_{J_t}. Without interval-wide existence, the statement holds on the corresponding flow domains.

Proof. Proposition A.3 gives $\partial_t F_{J_t} = \mathcal{F}_N(v_t) = R_t - DF_{J_t} U_t$. Hamilton's equations are

$$\dot{x} = U_t(x), \quad \dot{a} = 0, \quad \dot{\eta} = R_t(x), \quad \dot{\xi} = -DU_t(x)^\top \xi + DR_t(x)^\top a.$$

Along a spatial characteristic, set $\eta_t = F_{J_t}(x_t)$ and $\xi_t = DF_{J_t}(x_t)^\top a$. The chain rule verifies the η equation. Differentiating $\partial_t F_{J_t} = R_t - DF_{J_t} U_t$ in x and using the symmetry of second spatial derivatives verifies the ξ equation. Thus trajectories starting on the graph stay on it. The spatial flow is a diffeomorphism on the assumed interval; the remaining equations are linear or affine along it, so both forward and backward ambient trajectories exist. Applying the inverse flow proves equality of the transported graphs. Finally, Cartan's formula gives $\mathcal{L}_{X_{H_t}} \lambda = d(\lambda(X_{H_t}) - H_t)$. Integration in time shows that the ambient flow preserves λ up to an exact form and therefore preserves $\omega = -d\lambda$. □

Remark A.7 (Spatial discretization and the source field)**.** *For the DCT-based image velocity $v_t = r_\theta - (DF_{J_t}|_{\mathcal{G}_N})u_t$, the compatible source in equation 21 satisfies $R_t|_{\mathcal{G}_N} = r_\theta$. However, R_t need not equal $\mathcal{F}_N(r_\theta)$, since $DF_{J_t}U_t$ need not belong to $\mathcal{H}_N$.*

For video prediction, substituting $v_s = r_s^{\text{net}} + A(w_s)J_s$ and $u_s = w_s/N$ into equation 21 gives

$$r_s := R_s|_{\mathcal{G}_N} = r_s^{\text{net}} + A(w_s)J_s + \big(DF_{J_s}|_{\mathcal{G}_N}\big)u_s. \tag{22}$$

Thus r_s generally differs from r_s^{net} because the upwind and DCT transport terms differ. This correction specifies the compatible continuous Hamiltonian source; it is not added to the numerical update.

For the video variant, the predicted fields are frozen within each half-frame interval and may change discontinuously at interval boundaries. Proposition A.6 therefore applies intervalwise to the exact solutions of the frozen-field grid ODEs, subject to its spatial-flow existence assumptions. Since the image state is continuous across interval boundaries, the corresponding Hamiltonian flows can be composed to transport the image graphs over successive intervals. The implemented truncated-series updates approximate these exact grid evolutions.

Remark A.8 (Nonuniqueness and numerical interpretation)**.** *For the DCT velocity, $(u, r) \mapsto (u + \delta u, r + (DF_J|_{\mathcal{G}_N})\delta u)$ leaves v unchanged whenever the modified fields remain admissible. Thus the CFM objective alone does not uniquely identify the transport and source fields.*

B SUPPLEMENTARY EXPERIMENTS

We provide architecture, training, and evaluation details for image generation and deterministic video prediction. LHFM-I, LHFM-V, and the matched I-CFM baseline are trained from random initialization, without pretrained components, distillation, or additional fine-tuning.

B.1 IMAGE-GENERATION ARCHITECTURES AND HYPERPARAMETERS

The matched CIFAR-10 models share a time-dependent U-Net backbone and training settings (Table 4). Standard I-CFM predicts three image-velocity channels; LHFM-I predicts two transport and three source channels, adding only 2,306 parameters. LHFM-I uses $u_\theta = 0.125t \tanh(\widetilde{u}_\theta)$ and $v_\theta = r_\theta - (DF_J|_{\mathcal{G}_N})u_\theta$. Here $\widetilde{u}_\theta$ is the raw two-component network output. The elementwise nonlinearity bounds each grid component by $0.125t$ and suppresses transport near the noise endpoint. This is one choice allowed by Section 4.3. Both models use independent endpoint sampling with $\sigma = 0$ and the conditional flow-matching objective in Algorithm 1, with $v_\theta = f_\theta$ for I-CFM. The I-CFM baseline follows Tong et al. (2024) and the authors' TorchCFM method, using our local matched backbone rather than the upstream TorchCFM U-Net or a pretrained checkpoint; it is not an LHFM-I variant or minibatch OT-CFM.

Setting	**Standard I-CFM**	**LHFM-I**
Dataset / resolution	CIFAR-10 / 32×32 RGB	
Conditioning	Unconditional	
Base channels / multipliers	128 / $(1, 2, 2, 2)$	
Residual blocks per level (down / up)	$2/3$	
Attention resolutions	$16, 8$; 4 (bottleneck)	
Attention heads / channels per head	$4/64$	
Dropout	0.1	
Output channels	3 (velocity)	5 (transport and source)
Parameters	39,625,603	39,627,909
Batch size / training seed	$256/270829$	
Optimizer / coefficients / weight decay	AdamW / $(0.9, 0.999)$ / 0	
Peak / final scheduled learning rate	$2.5 \times 10^{-4}/2 \times 10^{-5}$	
Schedule / warmup	Cosine decay / 2,000 updates	
evaluated checkpoint	150k updates	
EMA target decay / gradient clipping	$0.9999/1$	
Network / field precision	BF16 / FP32	
Data augmentation	Random horizontal flip	
Training hardware	One NVIDIA GeForce RTX 5090	

Table 4: Settings for the matched CIFAR-10 comparison. Shared entries apply to both models; reported results use the 150k-update checkpoints.

Both models are initialized from scratch and sample training images uniformly with replacement. The EMA decay at update k is $\min\{0.9999, (1 + k)/(10 + k)\}$.

B.2 VIDEO-PREDICTION ARCHITECTURES AND HYPERPARAMETERS

Architecture. In LHFM-V, a convolutional encoder processes the observed frames, consecutive frame differences, and spatial coordinates. Temporal attention combines the observed history with six recent predicted states, while a separate convolutional module predicts features for all ten future frames. Separate heads with pixel-shuffle readouts produce the source r^{net} and transport w, without amplitude bounds or temporal gates. Predicted images are fed back into the recurrent state at each half-frame step; training uses the full rollout without teacher forcing. Figure 6 illustrates the architecture, and Table 5 summarizes its settings.

Training. We train from scratch using AdamW on a single NVIDIA GeForce RTX 5090. For the first 400,000 optimizer updates, we use a two-phase OneCycle schedule configured for 1,250,000

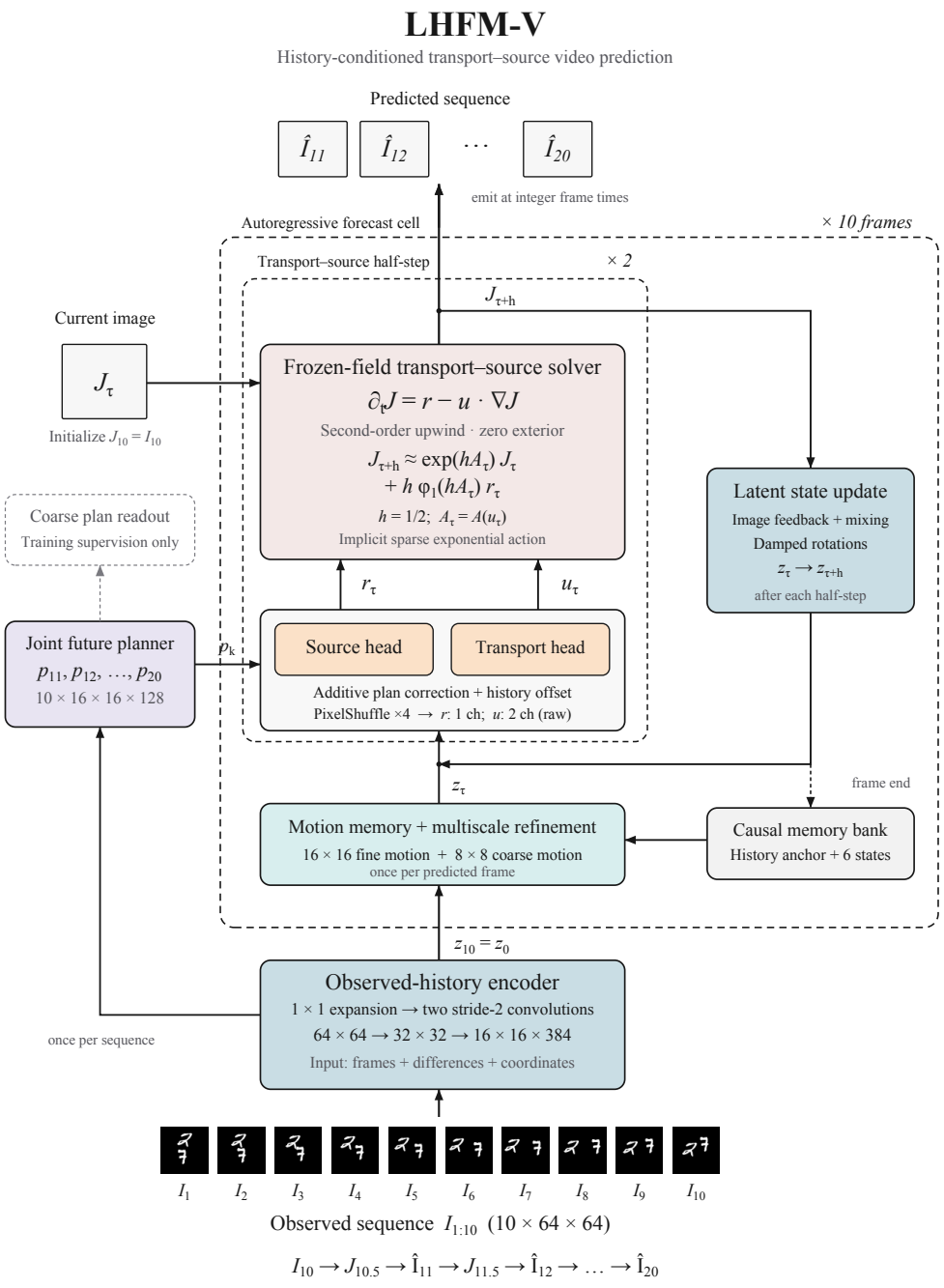

Figure 6: Architecture of LHFM-V for deterministic video prediction. Each future frame is produced by two transport–source half-steps, with image feedback and recurrent-state updates. In the diagram, τ, u, and r correspond to s, w, and r^{net} in the text, respectively; w is measured in pixels per frame and satisfies $w = Nu$ in the text's notation. The function $\varphi_1(X) = \int_0^1 e^{(1-\sigma)X}\, d\sigma$ represents the source integral in equation 12.

updates with cosine annealing and $\mathrm{pct_start} = 0.3$. The learning rate increases from 4×10^{-5} to 10^{-3} at update 375,000. We then resume from the complete 400,000-update checkpoint, preserving the model, optimizer, EMA, and random states. At update 400,001, we halve the learning rate prescribed by the original schedule to approximately 4.989935×10^{-4}, then apply cosine decay to 4×10^{-9} at update 1,250,000, without additional warmup. Adam's β_1 retains the original OneCycle trajectory, decreasing from 0.95 to 0.85 during warmup and subsequently increasing to 0.95; $\beta_2 = 0.999$. Neural training uses BF16, while image integration, recurrent states, and optimization use FP32. EMA evaluation uses FP32 throughout.

Setting	**LHFM-V**
Resolution / channels	64×64 / 1
Observed / future frames	$10/10$
Encoder widths / residual blocks	$(64, 128, 448)$ / 3
Recurrent-state channels / resolution	384 / 16×16
Memory frames / attention heads	6 plus history / 4
Memory key / value dimensions	$128/256$
Fine / coarse spatial channels	$320/192$
Future-feature channels / blocks	$128/4$
Source / transport head hidden width	$192/192$
Field evaluations per future frame	2
Parameters (including auxiliary head)	18,637,493
Batch size / training seed	$16/270829$
Optimizer / weight decay	AdamW / 10^{-4}
Adam β_1 range / β_2	$[0.85, 0.95]$ / 0.999
Initial / peak learning rate	$4 \times 10^{-5}/10^{-3}$
Updates at evaluation / learning rate	1.25M / 4×10^{-9}
EMA decay / gradient clipping (norm)	$0.999/1$

Table 5: Architecture and training settings for LHFM-V on Moving MNIST.

Objective. Let MSE denote the mean squared error over the minibatch, channels, and spatial positions. With $K = 10$ and two half-steps per future frame, the objective is

$$\begin{aligned}\mathcal{L}_{\mathrm{pred}} = \frac{1}{K}\sum_{k=1}^{K} \mathrm{MSE}(\widehat{I}_{T_{\mathrm{obs}}+k}, I_{T_{\mathrm{obs}}+k}) \\ + \frac{\lambda_r}{2K}\sum_{m=0}^{2K-1} \mathrm{mean}[(r_m^{\mathrm{net}})^2] + \frac{\lambda_u}{2K}\sum_{m=0}^{2K-1} \mathrm{TV}(w_m) \\ + \frac{\lambda_c}{K}\sum_{k=1}^{K} \mathrm{MSE}(\widehat{I}_k^{\mathrm{coarse}}, \mathcal{P}_4 I_{T_{\mathrm{obs}}+k}),\end{aligned} \tag{23}$$

where $\lambda_r = 10^{-3}$, $\lambda_u = 10^{-4}$, and $\lambda_c = 0.05$. Here $\mathcal{P}_4$ is average pooling with kernel and stride 4. A shared linear head produces the 16×16 auxiliary predictions $\widehat{I}_k^{\mathrm{coarse}}$ for training only; they do not enter the image rollout. The transport regularizer is

$$\mathrm{TV}(w) = \tfrac{1}{2}\left(\mathrm{mean}\,|w_{\cdot,j+1,\ell} - w_{\cdot,j,\ell}| + \mathrm{mean}\,|w_{\cdot,j,\ell+1} - w_{\cdot,j,\ell}|\right),$$

with means over the minibatch, both components, and valid adjacent pairs.

B.3 Numerical Implementation

Image generation. We evaluate the full-band DCT spatial Jacobian on the pixel grid and integrate $\dot{J}_t = v_\theta(t, J_t)$ with 128 Heun steps (256 NFE). The Hamiltonian is not explicitly evaluated during sampling.

Video prediction. Each future frame uses two half-frame transport–source updates. In normalized coordinates, the grid spacing is $\Delta = 1/N$; the implemented transport output $w = Nu$ uses pixels

per frame. Writing $w_i^+ = \max(w_i, 0)$ and $w_i^- = \max(-w_i, 0)$, the stencil is

$$[A(w)J](g) = \sum_{i=1}^{2} \Big\{ w_i^+(g) \left[-\tfrac{3}{2}J(g) + 2J(g - \Delta e_i) - \tfrac{1}{2}J(g - 2\Delta e_i) \right] + w_i^-(g) \left[-\tfrac{3}{2}J(g) + 2J(g + \Delta e_i) - \tfrac{1}{2}J(g + 2\Delta e_i) \right] \Big\}. \tag{24}$$

We use zero values outside the grid. This second-order upwind stencil discretizes $-U \cdot \nabla F$, not $-\nabla \cdot (UF)$.

With fields frozen over a half-step of length h, we set

$$q = \frac{3}{2} \max \left\{ 1, \max_{b,g} \big(|w_{b,1}(g)| + |w_{b,2}(g)| \big) \right\}, \qquad L = \max\{1, \lceil hq/3 \rceil\},$$

where b indexes the minibatch and g the spatial grid. Each internal interval has length $\delta = h/L$. Writing $P = \mathrm{Id} + A(w)/q$ and $\mu = q\delta$, we compute

$$z_0 = J, \qquad z_{k+1} = P z_k + r^{\mathrm{net}}/q, \qquad k = 0, \ldots, 23,$$

and update

$$J_{\mathrm{next}} = e^{-\mu} \sum_{k=0}^{24} \frac{\mu^k}{k!} z_k.$$

We repeat this update over the L internal intervals using the same predicted fields. The numerical scaling q and subdivision count L are treated as constants during differentiation; gradients propagate through $A(w)$ and r^{net}.

B.4 Evaluation Protocols

Image generation. We evaluate the 150k-update EMA checkpoints using 50,000 generated images and all 50,000 CIFAR-10 training images as the reference. Sampling uses Heun128 (256 NFE) with batch size 64. FID is computed with TorchMetrics 1.9.0 and torch-fidelity 0.4.0. Generated pixels are mapped from $[-1, 1]$ to $[0, 1]$ and clipped only for evaluation, with no intermediate solver clipping. The published entries in Table 2 use adaptive DOPRI5, unlike our fixed-step Heun evaluations. Published baselines retain their original protocols and are not matched-control comparisons.

Video prediction. Moving MNIST frames are normalized to $[0, 1]$. We partition the 60,000 MNIST training digit images into 55,000 training digits and 5,000 validation digits using split seed 271100. Training sequences are generated on demand from the training digit pool, with each sequence determined by data seed 270829 and its absolute sequence index. Each sequence contains two moving digits and 20 frames, split into ten observed and ten future frames. We construct 1,024 fixed validation sequences from the held-out digit pool using seed 271109 and select the EMA checkpoint with the lowest validation MSE. The results in Table 3 evaluate the selected 1.25M-step (2000 epochs) EMA checkpoint on the official 10,000-sequence test set, which was not used for checkpoint selection. MSE and MAE are the spatial sums of squared and absolute errors, respectively, on unclipped predictions, averaged over future frames and sequences. SSIM and PSNR use predictions clipped to $[0, 1]$. SSIM is computed using the scikit-image 0.19.3 implementation with a 7×7 uniform window, sample covariance, `data_range=2`, and $K_1 = 0.01$, $K_2 = 0.03$. The SSIM map is averaged over the interior after excluding a three-pixel border, then averaged equally over future frames and sequences. PSNR is $-10 \log_{10}(\max\{\mathrm{MSE}, 10^{-12}\})$ using mean-pixel MSE per frame, then averaged over frames and sequences. Each observed sequence produces one deterministic prediction.

C EXAMPLES

C.1 LHFM-I: IMAGE GENERATION

Figure 7 shows samples generated by LHFM-I on CIFAR-10.



Figure 7: CIFAR-10 samples from LHFM-I, generated using a 150k-update EMA checkpoint and 128 Heun steps (256 NFE). The 64-image grid is retained in its original order.

C.2 LHFM-V: VIDEO PREDICTION

Figures 8 and 9 show the first six official Moving MNIST test sequences in their original order (sequence IDs 0–5). Predictions use the same validation-selected 1.25M-update EMA checkpoint as Table 3.

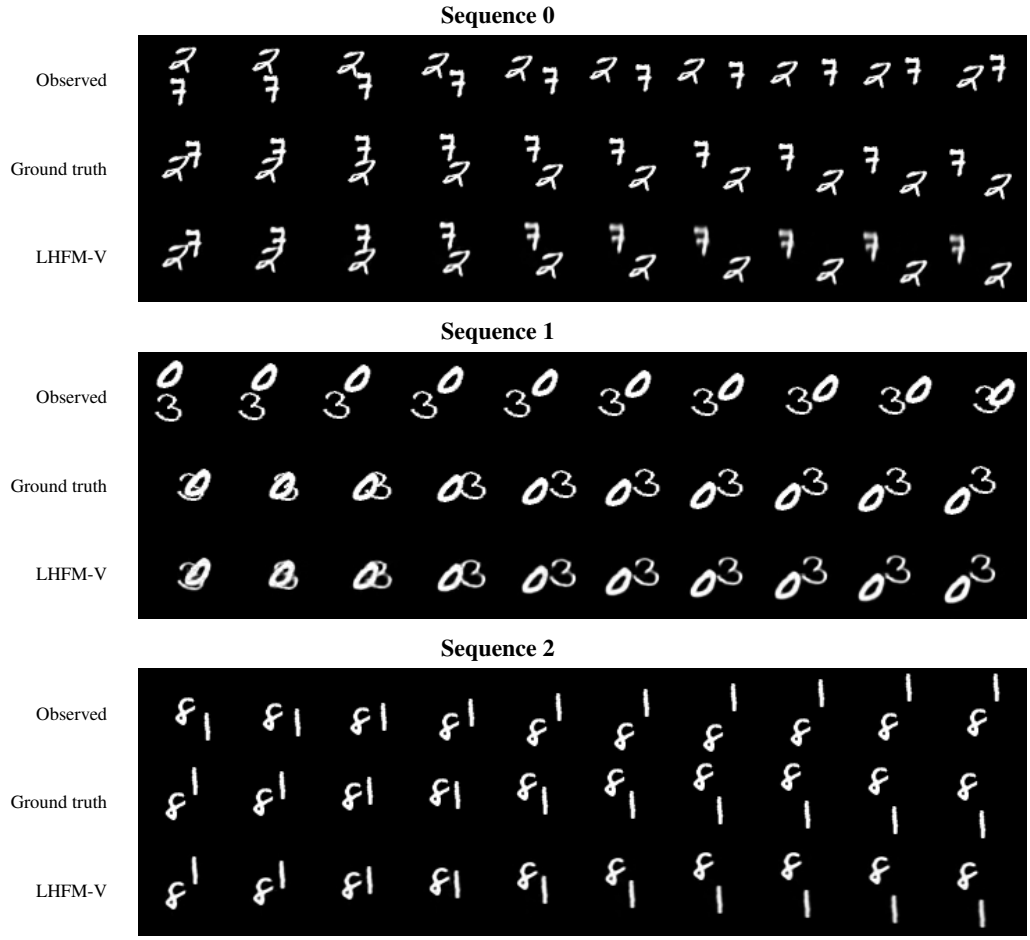

Figure 8: LHFM-V predictions on Moving MNIST (sequence IDs 0–2). For each sequence, rows show the ten observed frames (1–10), the ten ground-truth future frames (11–20), and the corresponding LHFM-V predictions. Time progresses from left to right.

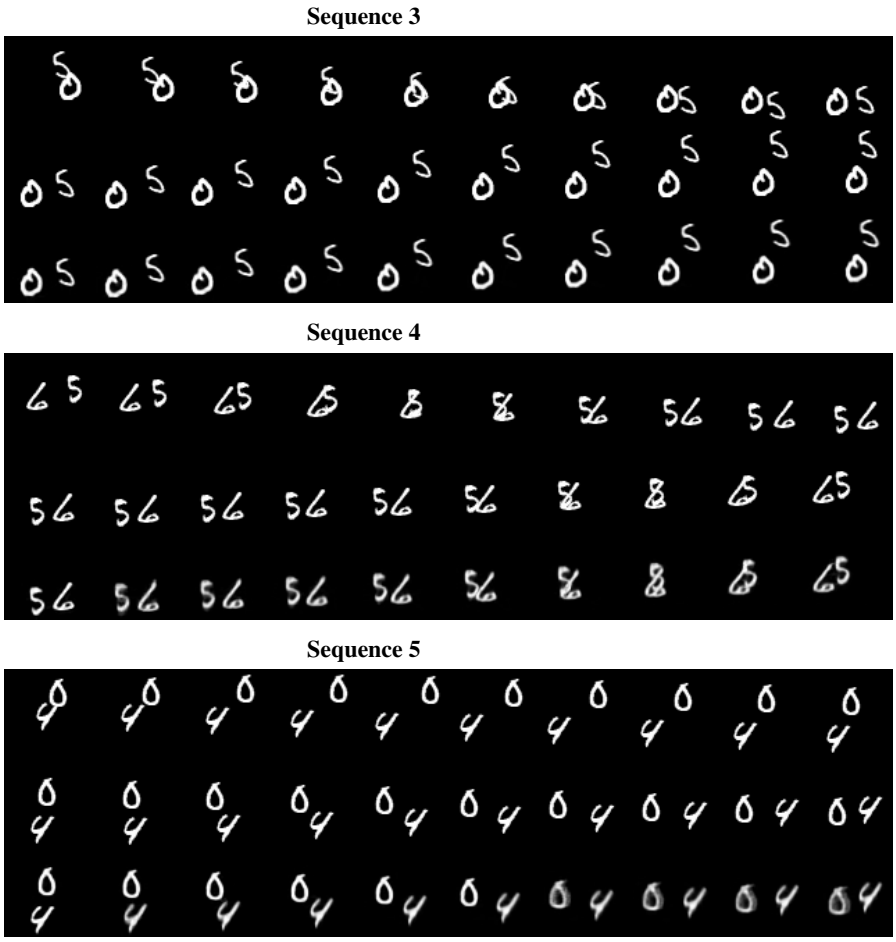

Figure 9: LHFM-V predictions on Moving MNIST (sequence IDs 3–5). Rows and time ordering follow Figure 8. Each observed sequence produces one deterministic prediction.