mathematics

MDPI

Article

Product Manifold Flow Matching: Low-NFE Generative Efficiency and Domain-Dependent Finite-Step Quality Optima

Hongseok Jang *, Youngtaek Cha and Sungjoon Choi

Advanced Mobility & Robot System Group, Daegyeong Technology Application Division, Korea Institute of Industrial Technology, Daegu 42994, Republic of Korea

* Correspondence: hsjang@kitech.re.kr; Tel.: +82-53-580-0323

Abstract

Generative flow quality need not improve monotonically as numerical integration is refined when the learned velocity field is itself approximate. We investigate this behavior in Product Manifold Flow Matching (PMFM), which represents image patches on Products of hyperbolic factors and samples with exponential-map Euler updates. Across 32 M and 129 M CIFAR-100 models, Product at NFE = 4 achieves lower FID than the original Euclidean baseline at NFE = 32 in all six architecture–training-seed configurations, with an approximately eightfold reduction in measured sampling time. A targeted 32 M ablation shows, however, that this crossover is not specific to Product geometry: a factor-wise Euclidean analog of the Product target-speed reweighting improves the Euclidean baseline across the evaluated NFE range and reproduces the cross-budget advantage in all three training seeds. Removing the same reweighting from the Product target degrades FID from NFE ≥ 4. Product FID reaches an early optimum at NFE = 8 on CIFAR-100 and at NFE = 4 on a balanced synthetic pile-driver corpus, while same-initial-condition step-doubling discrepancies continue to decrease. These results separate numerical refinement from generative fidelity and identify target-speed reweighting as a major contributor to low-NFE quality, without supporting a curvature-only causal interpretation.

Keywords: flow matching; product manifolds; hyperbolic geometry; numerical integration; generative models; sampling efficiency; construction imagery

MSC: 68T07; 65L05; 53B21

1. Introduction

Continuous generative flows transport a simple prior distribution to a data distribution along the solution of an ordinary differential equation (ODE) whose velocity field is learned from data. Flow matching trains a velocity field without repeatedly solving the model ODE during optimization [1], and generation is then carried out by numerically integrating the learned field from $t = 0$ to $t = 1$. The cost of generation is therefore governed in part by the number of function evaluations (NFE) used by the numerical solver. The interaction between the learned field and its discrete integrator is consequently not merely an implementation detail: it determines both computational budget and the distribution produced at finite step size.

For a sufficiently regular fixed vector field, a consistent numerical integrator converges to the ODE solution as the step size tends to zero under standard stability and regularity conditions [2]. However, asymptotic convergence neither guarantees monotonic

Check for updates

Academic Editor: Patricia J. Y. Wong

Received: 26 August 2026
Revised: 9 September 2026
Accepted: 10 September 2026
Published: 14 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

error reduction at finite integration budgets nor implies monotonic improvement in a distribution-level generative metric. A learned field is itself estimated from finite data under a finite training objective, so the trajectory approached by step refinement is the trajectory associated with the learned field rather than an idealized exact transport. There is therefore no a priori reason that the integration budget minimizing numerical discretization error must coincide with the budget minimizing a metric such as FID. Accordingly, whether finer integration improves sample quality for a learned generative flow is an empirical numerical question central to this study.

A second axis concerns the state space on which the learned dynamics are defined. Product Manifold Flow Matching (PMFM) represents each image patch by a Product of low-dimensional hyperbolic factors, uses geodesic interpolation to construct intermediate states, predicts tangent vectors on the Product manifold, and advances generated states through exponential-map Euler updates. The reference implementation used for the reported checkpoints also constructs its conditional training target from the initial geodesic tangent through an implementation-specific transport-and-projection operator. Section 3 defines this operator explicitly. Importantly, we do not identify it with Levi-Civita parallel transport; after projection, the resulting target is collinear with the geodesic tangent but has a time- and distance-dependent magnitude. The present study therefore evaluates the finite-step behavior of the reference implementation rather than claiming equivalence to canonical Riemannian flow matching [3].

We study numerical integration budget and Product-manifold state representation together using saved model checkpoints and controlled sampling protocols. The empirical evaluation yields four principal findings. First, the original Product system exhibits a favorable low-budget FID frontier: Product at NFE = 4 matches or outperforms the original Euclidean NFE = 32 reference across every evaluated domain/capacity–training-seed configuration. Second, a targeted component ablation shows that this cross-budget advantage is not uniquely attributable to the Product geometry. A factor-wise Euclidean analogue of the Product target-speed reweighting improves the Euclidean baseline throughout the evaluated NFE range and reproduces the NFE = 4 versus Euclidean@32 crossover in all three 32 M training seeds. Third, the best Product FID occurs at a finite integration budget rather than at maximal refinement. A same-initial-condition diagnostic shows successive discretizations continuing to approach one another even after FID has passed its optimum. Fourth, this finite-step behavior is not confined to a single image benchmark: the same 32 M Product architecture and manifold operators, with an exposure-matched training horizon, reproduce an early optimum on a structured synthetic construction-domain corpus, although its location shifts across domains.

To examine the third finding, we use a same-initial-condition step-refinement diagnostic. For a fixed prior realization, decoded samples are compared at successive NFE levels as the step count doubles. A monotone decrease in the step-doubling discrepancy provides empirical, Cauchy-like evidence that successive discrete solutions are refining toward one another, while distribution-level FID is tracked separately. This diagnostic separates numerical step refinement from distribution-level fidelity: the discrete trajectory refines, yet the quality metric has already passed its optimum. We do not claim convergence to an exact continuous-time solution, nor do we infer a causal mechanism for the mismatch.

The construction-domain experiment serves a deliberately restricted role. We do not use the pile-driver imagery to claim object detection, field deployment, or synthetic-to-real transfer. It provides a distinct structured image distribution on which the same 32 M model design can be re-evaluated without introducing a new latent encoder, resolution-specific architecture, or conditioning mechanism. Its training horizon is scaled to preserve approximately the same number of dataset passes as the CIFAR experiment. The pile

study is therefore an engineering-domain replication of the numerical behavior rather than evidence of deployment readiness.

The main contributions are:

1. Reproducible low-budget Product-versus-baseline ordering: Against the original Euclidean baseline, we establish a Product@4 < Euclidean@32 FID crossover across two model capacities and three training seeds per capacity, verify the ordering through independent checkpoint replay, and reproduce it in three additional construction-domain models—nine of nine evaluated domain/capacity–training-seed configurations.
2. Component-level attribution of target reweighting. A four-condition 32 M ablation separates target-speed reweighting from the original Product-versus-Euclidean comparison. A factor-wise Euclidean reweighting analog improves the Euclidean baseline at every evaluated NFE and reproduces the NFE = 4 versus Euclidean@32 crossover in all three training seeds, while removing the Product reweighting degrades Product FID from NFE ≥ 4. These results rule out attributing the low-NFE quality gain to manifold geometry alone.
3. Implementation-level geometric characterization. We derive a closed-form identity for the implemented Product target, showing that after destination tangent projection it is collinear with the geodesic tangent and differs only through the positive distance- and time-dependent factor $2\cosh(td)-1$ (Remark 1).
4. Finite-step quality optimum. Product FID reaches an early optimum at NFE = 8 on CIFAR in all six architecture–training-seed runs and at NFE = 4 on the construction corpus in all three training seeds. Thus, the existence of an early optimum generalizes while its numerical location is domain-dependent.
5. Refinement–fidelity decoupling. Under same-initial-condition diagnostics, step-doubling LPIPS decreases monotonically while Product FID has already passed its optimum. This establishes an empirical separation between numerical step refinement and distribution-level fidelity without asserting exact continuous-time convergence.
6. Construction-domain replication. We transfer the same 32 M architecture and manifold operators to a balanced 4000-image synthetic pile-driver corpus, with an exposure-matched training horizon, and observe lower Product FID in all 24 and lower Product KID in all 24 matched-NFE comparisons.

The remainder of the paper is organized as follows. Section 2 reviews flow matching, few-step generative sampling, and generative modeling on non-Euclidean spaces. Section 3 gives the explicit Product-manifold operators, the implementation-specific target construction, the two samplers, and the step-refinement diagnostic. Section 4 reports benchmark and construction-domain results. Section 5 discusses interpretation and limitations, and Section 6 concludes.

2. Related Work

2.1. Flow Matching and Low-Cost Continuous Generative Sampling

Flow Matching (FM) provides a simulation-free training framework for continuous normalizing flows by regressing conditional vector fields along prescribed probability paths [1]. Subsequent methods have explicitly targeted flows that can be sampled accurately with few function evaluations. Rectified Flow seeks straighter transport paths so that coarse numerical discretization remains effective [4]. Multisample Flow Matching uses minibatch couplings to reduce gradient variance and obtain straighter learned flows [5], while generalized Conditional Flow Matching and OT-CFM use alternative couplings, including minibatch optimal transport, to simplify transport and improve inference efficiency [6]. Few-step generation is also pursued outside flow matching; consistency models, for example, modify the training target to permit one- or few-step generation [7].

These approaches establish low-step generation as an important computational objective, but they primarily change path construction, coupling, distillation, or the learning objective. The present work instead studies how generative quality changes under step refinement for a fixed trained Product-manifold implementation, and whether numerical refinement and distribution-level fidelity behave monotonically together.

2.2. Generative Modeling on Non-Euclidean Spaces

Non-Euclidean representation learning has motivated neural models whose latent variables and operations are defined on manifolds. Hyperbolic neural networks extend standard neural operations to hyperbolic geometry [8], Poincaré variational autoencoders use hyperbolic latent variables for generative representation learning [9], and Riemannian score-based generative modeling generalizes score methods to manifold-valued data [10].

The closest conceptual precedent is Riemannian flow matching, which formulates flow matching on general geometries through manifold-aware conditional vector fields [3]. The evaluated PMFM system in this study likewise uses a native Product-hyperbolic state representation, geodesic interpolation, tangent projection, and exponential-map sampling. However, its authoritative training implementation uses the explicit transport-and-projection target stated in Section 3 rather than the canonical Levi-Civita Riemannian flow-matching target. We therefore compare the implemented system empirically and do not claim theoretical equivalence to canonical Riemannian flow matching.

2.3. Synthetic Imagery in Construction Computer Vision

Construction vision is frequently constrained by the cost of collecting and annotating diverse site imagery. Synthetic data has therefore become an established strategy for expanding construction datasets and controlling scene variability [11]. Prior work has developed synthetic construction-equipment datasets under controlled backgrounds, poses, and viewpoints [12], large automated construction-image synthesis and labeling frameworks [13], hybrid real–synthetic datasets for tool recognition [14], and site-specific synthetic-data pipelines for equipment detection [15].

The construction corpus in the present study serves a more limited purpose. It is not used to claim improved detection, field deployment, or synthetic-to-real transfer. Instead, it provides a structured engineering-domain distribution on which the same 32 M Product/Euclidean model design can be retrained and the finite-step behavior independently re-evaluated.

3. Materials and Methods

3.1. Patch-Space Product-Manifold State

Each 32×32 RGB image x is partitioned into $P = 64$ non-overlapping 4×4 patches. Each patch is a vector in $\mathbb{R}^{48}$. For the Product model, a patch vector

$$u = (u_1, \ldots, u_{48}) \in \mathbb{R}^{48} \tag{1}$$

is split into $m = 24$ ordered pairs $u^{(j)} \in \mathbb{R}^2$, and each pair is lifted to one factor of the Lorentz model of $\mathbb{H}^2$. The per-patch state space is

$$\mathcal{M} = \prod_{j=1}^{24} \mathbb{H}^2, \qquad \dim\mathcal{M} = 48 \tag{2}$$

Across all patches, the image-level state space is

$$\mathcal{X} = \mathcal{M}^{64} = \left(\mathbb{H}^2\right)^{1536}, \qquad \dim\mathcal{X} = 3072 \tag{3}$$

represented by $1536 \times 3 = 4608$ ambient Lorentz coordinates.

The representation, target construction, and sampling workflow is summarized in Figure 1.

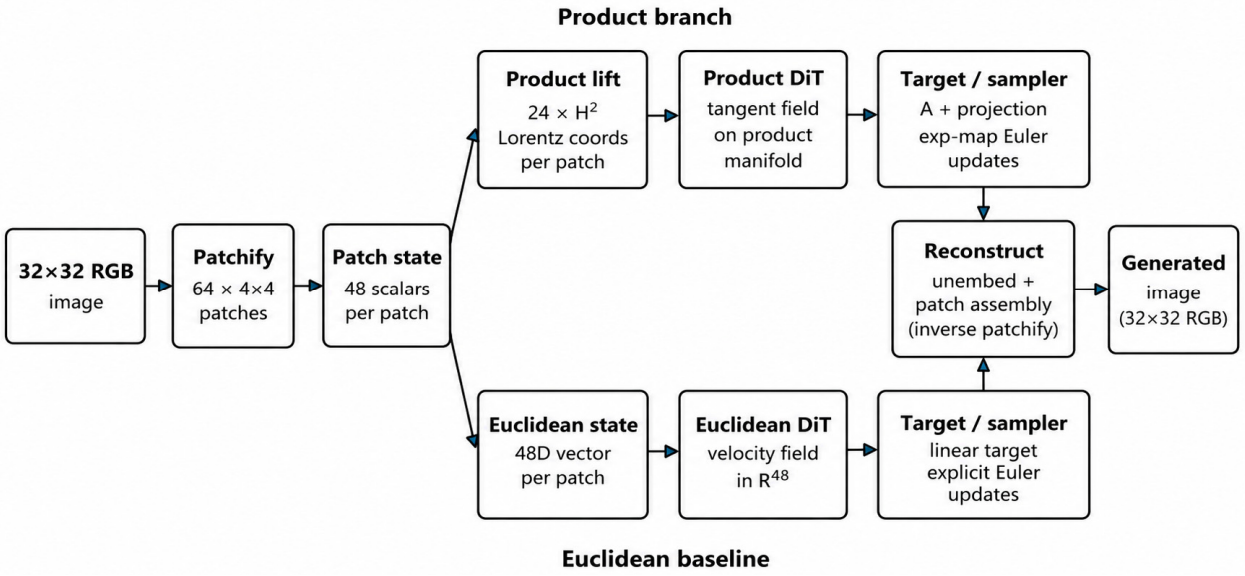


Figure 1. Evaluated Product Manifold Flow Matching implementation and Euclidean baseline. Image patches are represented either as 24 two-dimensional hyperbolic factors or as 48-dimensional Euclidean patch states. The Product training target uses the implementation transport operator followed by destination tangent projection; this operator is not identified with Levi-Civita parallel transport. Product sampling uses exponential-map Euler updates, whereas the baseline uses explicit Euler updates in Euclidean patch space.

3.2. Lorentz Model of $\mathbb{H}^2$

For $a, b \in \mathbb{R}^3$, define the Lorentz inner Product

$$\langle a, b \rangle_{\mathcal{L}} = -a_0 b_0 + a_1 b_1 + a_2 b_2 \tag{4}$$

A point $z = (z_0, z_1, z_2)$ lies on the unit-curvature upper hyperboloid when

$$\langle z, z \rangle_{\mathcal{L}} = -1, \qquad z_0 > 0 \tag{5}$$

The implementation enforces this constraint by replacing the time coordinate while preserving the two spatial coordinates:

$$\mathrm{norm}(z) = \left(\sqrt{1 + z_1^2 + z_2^2}, z_1, z_2 \right) \tag{6}$$

The tangent space is

$$T_z \mathbb{H}^2 = \left\{ v \in \mathbb{R}^3 : \langle z, v \rangle_{\mathcal{L}} = 0 \right\} \tag{7}$$

and for a tangent vector

$$\| v \|_{\mathcal{L}}^2 = \left\langle v, v \right\rangle_{\mathcal{L}} = -v_0^2 + v_1^2 + v_2^2 \geq 0 \tag{8}$$

All operators act independently factor-wise across the Product.

3.3. Exponential Map, Tangent Projection, and Deterministic Lift

For $z \in \mathbb{H}^2$ and $v \in T_z\mathbb{H}^2$,

$$\exp_z(v) = \cosh(\| v \|_{\mathcal{L}})z + \sinh(\| v \|_{\mathcal{L}})\frac{v}{\| v \|_{\mathcal{L}}} \tag{9}$$

At zero tangent norm, (9) is interpreted continuously as $\exp_z(0) = z$. The implementation uses small positive clamps in the norm and inverse-hyperbolic-cosine calculations to avoid numerical division by zero.

An arbitrary ambient vector $w \in \mathbb{R}^3$ is projected onto $T_z\mathbb{H}^2$ by

$$\mathrm{proj}_z(w) = w + \langle z, w \rangle_{\mathcal{L}} z \tag{10}$$

For a scaled coordinate pair $u^{(j)} = (u_1, u_2) \in \mathbb{R}^2$, the deterministic lift is

$$\iota\left(u^{(j)}\right) = \mathrm{norm}(0, u_1, u_2) = \left(\sqrt{1 + u_1^2 + u_2^2}, u_1, u_2\right) \tag{11}$$

The inverse unembed is the exact spatial projection

$$\iota^{-1}(z) = (z_1, z_2) \tag{12}$$

Before lifting, image-patch coordinates are divided by SCALE_FACTOR $= 4.0$; after unembedding, they are multiplied by the same factor.

3.4. Logarithmic Map and Implementation Transport Operator

For $z, y \in \mathbb{H}^2$, let

$$d(z, y) = \mathrm{arcosh}\left(-\langle z, y \rangle_{\mathcal{L}}\right) \tag{13}$$

The logarithmic map used in the implementation is

$$\log_z(y) = d(z, y)\frac{y + \langle z, y \rangle_{\mathcal{L}} z}{\| y + \langle z, y \rangle_{\mathcal{L}} z \|_{\mathcal{L}}} \tag{14}$$

The code-level transport routine is represented by the neutral operator

$$\mathcal{A}_{x \to y}(v) = v + \langle x, v \rangle_{\mathcal{L}} y - \frac{\langle y, v \rangle_{\mathcal{L}}}{1 - \langle x, y \rangle_{\mathcal{L}}}(x + y) \tag{15}$$

For tangent $v \in T_x\mathbb{H}^2$, the middle term vanishes. The implementation transport operator is not asserted to be Levi-Civita parallel transport: in general, $\mathcal{A}_{x \to y}(v) \notin T_y\mathbb{H}^2$. The reference implementation therefore applies an explicit tangent projection at the destination after this operator.

The geodesic interpolation used to construct intermediate Product states is

$$\gamma_x^{z_1}(t) = \exp_x\left(t \log_x(z_1)\right), \qquad t \in [0, 1] \tag{16}$$

3.5. Priors and Interpolation

The Product prior is sampled factor-wise around the origin $o = (1, 0, 0)$. An ambient Gaussian draw $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_3)$ is tangent-projected at o, mapped by the exponential map, and normalized:

$$z_0 = \mathrm{norm}[\exp_o\left(\mathrm{proj}_o(\varepsilon)\right)], \qquad \sigma = 0.15 \tag{17}$$

The Euclidean baseline uses $z_0 \sim \mathcal{N}(0, \sigma^2 I)$ in $\mathbb{R}^{48}$ per patch with the same σ.

For Product data state z_1 and $t \sim \mathcal{U}[0,1]$,

$$z_t = \text{norm}\left[\exp_{z_0}\left(t \log_{z_0}(z_1)\right)\right] \tag{18}$$

3.6. Conditional Target Construction and Losses

Let

$$v_0 = \log_{z_0}(z_1) \tag{19}$$

The Product target used in the reference implementation is

$$v^\star(z_t) = \text{proj}_{z_t}[\mathcal{A}_{z_0 \to z_t}(v_0)] \tag{20}$$

This target reproduces the implementation order

$$\log_{z_0}(z_1) \;\to\; \mathcal{A}_{z_0 \to z_t} \;\to\; \text{proj}_{z_t} \tag{21}$$

Remark 1 (direction-preserving target reweighting). Let $d = d(z_0, z_1)$, and let

$$u = \frac{\log_{z_0}(z_1)}{d} \tag{22}$$

By construction, u is a unit tangent vector at z_0: it is Lorentz-orthogonal to z_0 and has unit Lorentz norm. The geodesic can then be written as

$$\gamma(t) = \cosh(td) z_0 + \sinh(td) u \tag{23}$$

For $z_t = \gamma(t)$, direct substitution into the implementation operator followed by destination tangent projection gives

$$v^\star(\gamma(t)) = [2\cosh(td) - 1]\dot{\gamma}(t) \tag{24}$$

Hence, the implemented target is collinear with the geodesic tangent and preserves its direction, but its magnitude is reweighted by a positive time- and pair-distance-dependent factor. Except in limiting or special cases, it is not identical to $\dot{\gamma}(t)$. We therefore report the implementation exactly and do not claim that this target is the canonical Levi-Civita Riemannian flow-matching target.

The Product network output is projected onto the tangent space at z_t. Let

$$r_\theta = v_\theta(z_t, t) - v^\star(z_t) \tag{25}$$

The Product loss used in the implementation is

$$\mathcal{L}_{\text{prod}}(\theta) = \mathbb{E}_{z_0, z_1, t}[\max\{0, \langle r_\theta, r_\theta \rangle_{\mathcal{L}}\}] \tag{26}$$

averaged over factors, patches, and batch elements.

For the Euclidean baseline,

$$z_t = (1-t) z_0 + t z_1 \tag{27}$$

with constant target $z_1 - z_0$, and

$$\mathcal{L}_{\text{euc}}(\theta) = \mathbb{E}_{z_0, z_1, t}\left[\| v_\theta(z_t, t) - (z_1 - z_0) \|_2^2\right] \tag{28}$$

At a fixed capacity, Product and Euclidean variants share the transformer backbone but use representation-specific input/output projections: 72 ambient Lorentz coordinates versus 48 Euclidean patch coordinates. They are therefore near-capacity-matched rather than parameter-identical.

3.7. Target-Reweighting Attribution Ablation

To address the fact that the original Product–Euclidean comparison jointly changes the state representation, prior, conditional target, and sampling update, we performed a targeted four-condition ablation at the 32 M capacity. The original Euclidean condition (E0) uses the Gaussian prior, constant linear target, and explicit Euler sampler. The reweighted Euclidean condition (E1) retains the same representation, prior, architecture, and sampler as E0, but replaces the constant target with a factor-wise Euclidean analog of the Product target-speed reweighting.

To preserve the factor structure of the Product representation, each 48-dimensional Euclidean patch displacement is reshaped into 24 two-dimensional pairs. For pair j with Euclidean displacement magnitude d_j^E the E1 target is multiplied by

$$\omega_j^E(t) = 2\cosh\left(t\, d_j^E\right) - 1 \tag{29}$$

This construction is a structural Euclidean analog of the Product reweighting in Equation (20), rather than a literal application of the hyperbolic expression in Euclidean space.

The unreweighted Product condition (P1) retains the Product representation, wrapped-normal prior, network architecture, tangent projection, and exponential-map Euler sampler of the original Product condition (P0), but removes the positive target-speed factor $2\cosh(t\,d) - 1$. P1 therefore uses the corresponding constant-speed geodesic tangent target. The contrasts E0→E1 and P1→P0 isolate the effect of target-speed reweighting within the Euclidean and Product formulations, respectively. In contrast, E1 versus P0 does not isolate manifold geometry because the representation, prior, and sampling update remain different.

E1 and P1 were trained using the same 32 M DiT-S configuration as the corresponding original models: 80,000 optimization steps, batch size 128, and independent training seeds 42, 123, and 456. Each of the four conditions was evaluated with 5000 generated samples at NFE = 1, 2, 4, 8, 16, and 32. Training seeds are treated as the independent model units.

3.8. Sampling

For K uniform steps, $h = 1/K$. The Euclidean sampler uses explicit Euler:

$$z^{(k+1)} = z^{(k)} + h\, v_\theta\left(z^{(k)}, kh\right) \tag{30}$$

The Product sampler uses exponential-map Euler followed by normalization:

$$z^{(k+1)} = \mathrm{norm}\left[\exp_{z^{(k)}}\left(h\, v_\theta\left(z^{(k)}, kh\right)\right)\right] \tag{31}$$

The reference sampler uses no velocity clipping, state reset, or geometry-specific rescue rule. Product states are unembedded through ι^{-1}, multiplied by SCALE_FACTOR $= 4.0$, and rearranged into 64 patches. No autoencoder decoder is used.

3.9. Same-Initial-Condition Step-Refinement Diagnostic

For each geometry, a fixed set of prior samples is reused across the NFE sweep. Let $G_K^{(i)}$ denote the decoded sample from initial condition i using K integration steps. Define

$$D(K, 2K) = \frac{1}{N} \sum_{i=1}^{N} \ell \left(G_K^{(i)}, G_{2K}^{(i)} \right) \tag{32}$$

with LPIPS as the primary discrepancy and pixel RMSE as a secondary measure. A monotone decrease in $D(K, 2K)$ is described only as empirical, Cauchy-like step refinement of successive discrete solutions. It is not a proof of convergence to an exact continuous-time solution. Product and Euclidean samples are not paired across geometries because their priors differ.

3.10. Evaluation Lineages and Statistical Units

Five evaluation lineages are kept distinct.

1. Creator CIFAR NFE spectra. These provide the primary benchmark FID-versus-NFE curves, the Product@4 versus Euclidean@32 headline comparisons, and the measured sampling-time ratios in Figure 2.
2. Target-reweighting attribution ablation. The 32 M E0/E1/P1/P0 experiment isolates target reweighting within each representation family. E1 and P1 are independently trained under seeds 42, 123, and 456, while E0 and P0 use the corresponding original 32 M checkpoints. Evaluation realizations are matched within the E0/E1 and P1/P0 comparisons at each training-seed–NFE condition. Different NFE values are not treated as same-initial-condition trajectories; finite-step refinement claims therefore continue to rely on the dedicated diagnostic lineage.
3. Independent checkpoint replay. Saved checkpoints are regenerated under new evaluation seeds to test robustness to Monte-Carlo generation noise. Evaluation seeds are nested repeats within a trained model and are not treated as independent model replicates.
4. Same-initial-condition refinement diagnostic. Figure 3 uses a fixed generation realization for each trained model (evaluation seed 9001) so that FID and step-doubling discrepancies refer to the same diagnostic realization. The overlapping 129 M conditions coincide numerically with the corresponding independent-replay realization at evaluation seed 9001; these diagnostic values remain distinct from the creator NFE spectra.
5. Pile-domain evaluation. Figures 4–6 report results from the fixed 4000-image pile corpus and the independently trained pile Product/Euclidean models. Figure 4 combines source-resolution corpus examples with quantitative holdout metrics, while Figures 5 and 6 show within-geometry fixed-initial-condition qualitative progressions. Absolute pile FID is not compared numerically with CIFAR FID.

Training seeds are the independent trained-model units. Generation/evaluation seeds are Monte-Carlo repeats nested within those trained models.

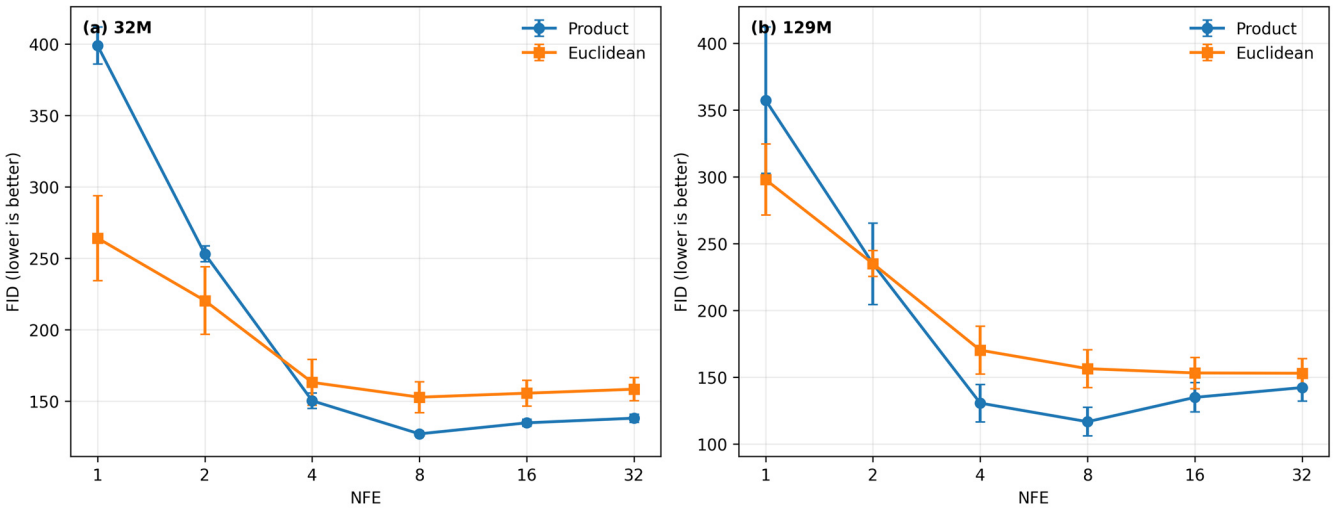


Figure 2. Creator-run CIFAR-100 FID versus NFE for (**a**) 32 M and (**b**) 129 M models. Points show the means across three training seeds and error bars show the between-seed standard deviation. Product FID reaches its minimum at NFE = 8 in all six architecture–training-seed configurations, and Product@4 is below Euclidean@32 in all six. Independent checkpoint replay is treated separately as robustness evidence.

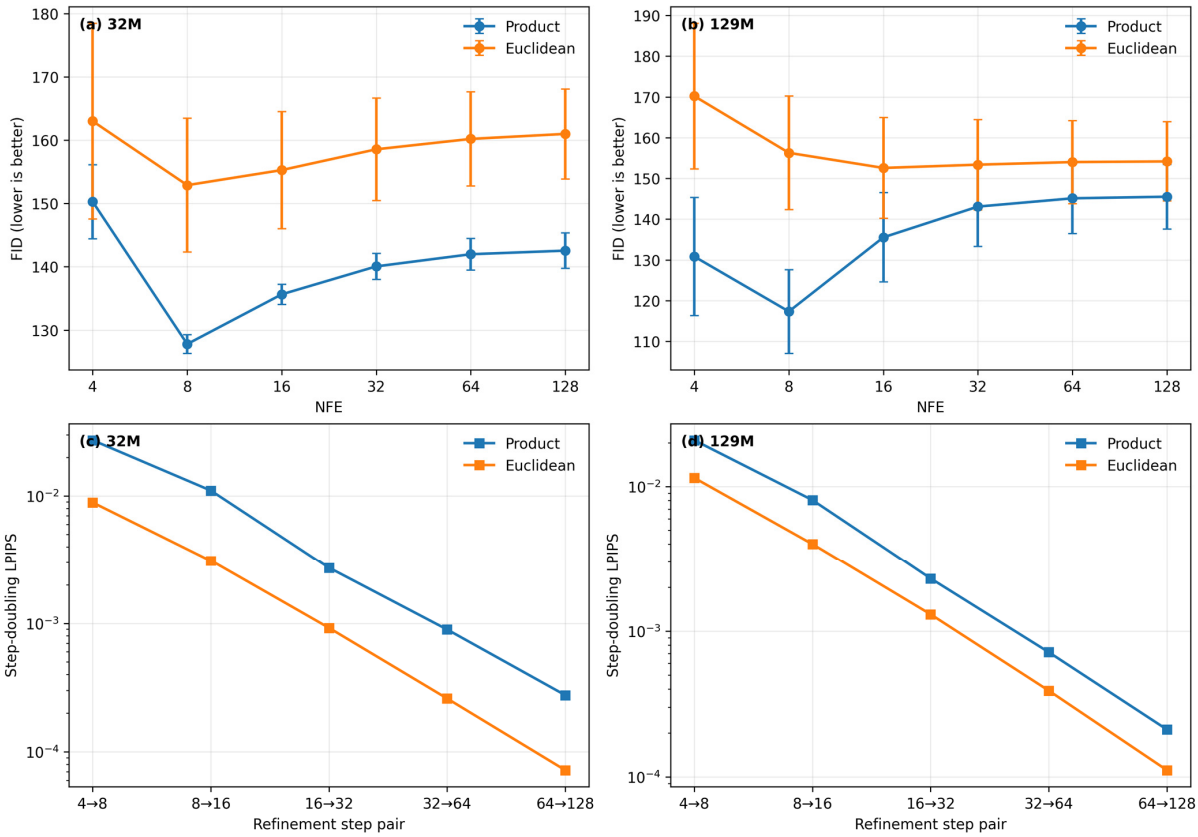


Figure 3. Numerical refinement versus generative fidelity on CIFAR-100. Results are from the same-initial-condition M1/M1-ext diagnostic runs using evaluation seed 9001 for each trained model; this diagnostic lineage is distinct from the creator spectra in Figure 2. Panels (**a**,**b**): FID reaches an early finite-step optimum and worsens under further refinement. Panels (**c**,**d**): step-doubling LPIPS decreases monotonically; the underlying seed-level trajectories are monotone in all 12 geometry–capacity–training-seed configurations. Error bars in the FID panels indicate variation across the three training seeds.

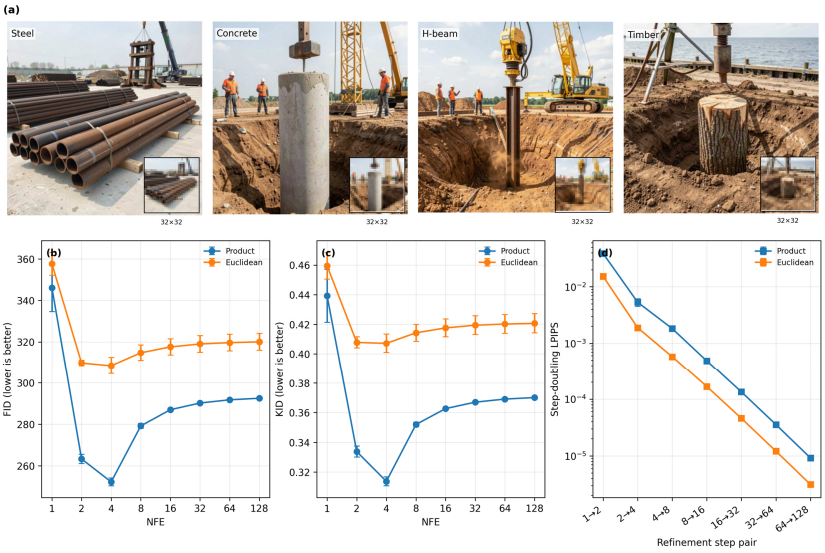


Figure 4. Structured synthetic pile-driver corpus and quantitative domain replication. (**a**) Representative 768 × 768 source images from the synthetic pile-driver corpus; the inset in each example is the corresponding 32 × 32 bicubic model input enlarged for visibility. These source-resolution examples distinguish corpus image quality from resolution-induced blur in the model inputs. (**b**) FID and (**c**) KID across NFE = 1–128; Product is lower in all 24 matched-NFE seed-level comparisons for each metric and Product FID is minimized at NFE = 4 in all three training seeds. (**d**) Step-doubling LPIPS decreases monotonically across all six pile geometry–training-seed trajectories.



Figure 5. Product fixed-initial-condition sampling progression on the synthetic pile-driver domain. Columns show NFE = 1, 2, 4, 8, and 32. Each row holds the Product prior realization fixed across NFE, so changes within a row reflect numerical refinement of the same initial condition. The panel is qualitative only; distribution-level quality and the NFE = 4 Product optimum are assessed using FID and KID.

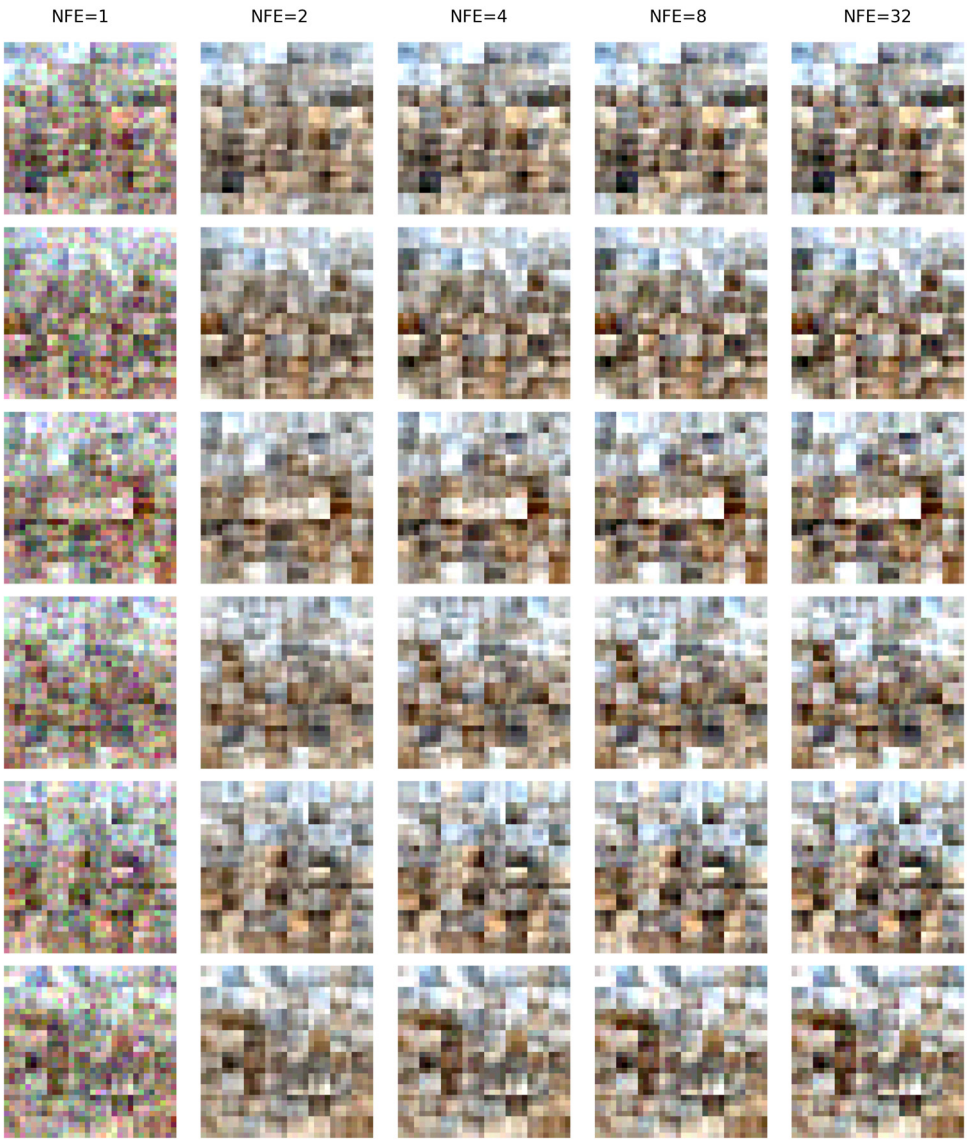


Figure 6. Euclidean fixed-initial-condition sampling progression on the synthetic pile-driver domain. Columns show NFE = 1, 2, 4, 8, and 32. Each row holds the Euclidean prior realization fixed across NFE. Product and Euclidean rows are displayed separately and are not interpreted as paired cross-geometry comparisons because the two geometries use different priors.

3.11. Notation Summary

The principal notation used in the mathematical formulation is summarized below.

Symbol	Meaning
$\mathbb{H}^2, \mathcal{M} = (\mathbb{H}^2)^{24}, \mathcal{X} = \mathcal{M}^{64}$	Hyperbolic factor, per-patch manifold, image-level Product manifold
$\langle \cdot, \cdot \rangle_{\mathcal{L}}$	Lorentz inner Product
$\mathrm{norm}, \iota, \iota^{-1}$	Hyperboloid constraint operator, lift, exact unembed
$\exp_z, \log_z$	Exponential and logarithmic maps
proj_z	Tangent projection
$\mathcal{A}_{x \to y}$	Implementation transport operator; not Levi-Civita PT
$v^\star, v_\theta$	Implemented Product target and learned tangent field
$K, h = 1/K$	NFE and integration step size
$D(K, 2K)$	Step-doubling discrepancy
$\sigma = 0.15$, SCALE_FACTOR $= 4.0$	Prior scale and patch rescaling

3.12. Experimental Setup

CIFAR-100 [16] is used as the benchmark image distribution. All images are represented at 32×32 resolution and partitioned into 64 non-overlapping 4×4 patches. Two transformer capacities are evaluated. The 32 M model uses hidden width 384, 12 transformer blocks, and 6 attention heads; the 129 M model uses hidden width 768, 12 blocks, and 12 heads. The architecture follows the transformer-based image-generation design used by diffusion transformers [17], adapted here to the Product and Euclidean state representations. Product and Euclidean variants share the same transformer backbone at a given capacity but use representation-specific input/output projections, yielding 32,315,592 versus 32,297,136 parameters at 32 M and 128,921,928 versus 128,885,040 parameters at 129 M.

Training uses batch size 128, learning rate 5×10^{-5}, Adam optimization [18], weight decay 10^{-4}, gradient clipping at 1.0, $\sigma = 0.15$, and cosine learning-rate decay. The original CIFAR models and the newly trained E1/P1 ablation variants used 80,000 optimization steps with training seeds 42, 123, and 456. The creator NFE spectrum evaluates $K \in \{1, 2, 4, 8, 16, 32\}$ with 5000 generated samples per condition. Independent checkpoint replay used additional evaluation seeds solely as nested Monte-Carlo repeats. The model architecture and optimization settings are summarized in Table 1.

Table 1. Model architecture and optimization settings. Product and Euclidean models are near-capacity-matched rather than parameter-identical.

Setting	CIFAR 32 M	CIFAR 129 M	Pile 32 M
Image input	32×32	32×32	32×32 (from 768×768 bicubic)
Patch size	4×4	4×4	4×4
Patches/image	64	64	64
Euclidean patch dimension	48	48	48
Product factors/patch	$24 \times \mathbb{H}^2$	$24 \times \mathbb{H}^2$	$24 \times \mathbb{H}^2$
Hidden width	384	768	384
Transformer blocks	12	12	12
Attention heads	6	12	6
Parameters—Euclidean	32,297,136	128,885,040	32,297,136
Parameters—Product	32,315,592	128,921,928	32,315,592
Batch size	128	128	128
σ	0.15	0.15	0.15
Optimizer	Adam	Adam	Adam
Weight decay	10^{-4}	10^{-4}	10^{-4}
Gradient clip	1.0	1.0	1.0
Training steps	80,000	80,000	5120
Training seeds	42, 123, 456	42, 123, 456	42, 123, 456
Primary NFE grid	1, 2, 4, 8, 16, 32	1, 2, 4, 8, 16, 32	1, 2, 4, 8, 16, 32, 64, 128

3.13. Structured Pile-Driver Corpus and Domain-Replication Protocol

The construction-domain replication uses 4000 unique synthetic pile-driver images organized as a balanced factorial design: four pile material classes and ten scene categories, with 100 unique images per class-category cell. The corpus was generated as a controlled cross-domain replication dataset rather than as a substitute for field-acquired data. Balancing material and scene factors provides a structured distribution that differs substantially from CIFAR-100 while allowing the same unconditional 32 M architecture and patch representation to be retained. The source images are 768×768 RGB, preserving the visual structure of construction scenes, whereas deterministic bicubic downscaling to 32×32 is used solely to match the benchmark architecture. Figure 4a therefore shows representative source-resolution images together with their corresponding 32×32 model inputs so that the resolution constraint is explicit. An inventory audit identified and removed nine redundant exact duplicates originating from notebook checkpoint folders and

found no corrupted images in the finalized corpus. The 4000 unique images are split within every class-category cell into 80 training and 20 holdout images, giving 3200 training and 800 holdout images.

A semantic-retention assessment at 32×32 yields four-way material leave-one-category-out accuracy 0.604 and macro-F1 0.558, compared with chance level 0.25, and ten-way scene-category accuracy 0.917. These values support the presence of coarse material and strong scene information after resizing but are not interpreted as engineering-grade recognition performance. Table 2 summarizes the corpus characterization.

Table 2. Structured synthetic pile-driver corpus characterization.

Item	Value
Unique images	4000
Material classes	4
Scene categories	10
Images per class-category cell	100
Train/holdout	3200/800
Source resolution	768×768 RGB
Model input	32×32 bicubic
4-way material LOCO accuracy	0.604
4-way material macro-F1	0.558
10-way scene-category accuracy	0.917
Exact duplicate excess removed	9
Corrupt images	0
Primary role	Structured synthetic construction-domain replication

Pile-domain training used the same 32 M Product and Euclidean architectures and optimization settings as the CIFAR 32 M experiment. The training horizon was scaled to 5120 steps so that the effective number of dataset passes matched that of the CIFAR 32 M experiment: $80,000 \times 128/50,000 = 5120 \times 128/3200 = 204.8$. The cosine schedule is scaled to the 5120-step horizon. Product and Euclidean models were independently trained using seeds 42, 123, and 456.

3.14. Evaluation Metrics

Fréchet Inception Distance (FID) [19] is the primary distribution-level quality measure. Kernel Inception Distance (KID) [20] is used as secondary evidence, particularly for the pile-domain replication. Same-initial-condition refinement is measured primarily with LPIPS [21] and secondarily with pixel RMSE. Because the pile holdout contains 800 images and represents a different distribution from CIFAR-100, absolute FID values are not compared across the two domains. Statistical summaries report variation across independently trained models; evaluation seeds are nested Monte-Carlo repeats and are not counted as independent training replicates. Table 3 maps these five lineages to their corresponding statistical units and manuscript evidence to prevent creator runs, checkpoint-replay realizations, refinement diagnostics, and independently trained pile-domain models from being interpreted as interchangeable replicates.

Table 3. Mapping of evaluation lineages, statistical units, and manuscript evidence.

Evaluation Lineage	Purpose	Training Seed/Unit	Evaluation Seed	Main Evidence
Creator CIFAR NFE spectra	Provide primary FID–NFE frontier and timing	42, 123, 456; trained models are independent units	Creator evaluation realization	Figure 2, Table 4
Independent checkpoint replay	Test robustness to generation Monte-Carlo noise	Same three trained 129 M models	9001, 9002, 9003 nested within each model	Section 4.4
Same-initial-condition refinement diagnostic	Separate numerical refinement from FID behavior	Same trained models	Fixed seed 9001; overlaps with replay at 129 M	Figure 3
Target-reweighting ablation	Component attribution	42, 123, 456; Independently trained model units	Matched within E0/E1 and P1/P0 at each NFE	Section 4.2; Table 5
Pile-domain evaluation	Independent engineering-domain replication	42, 123, 456 Independently trained pile models	Fixed protocol per evaluation	Figures 4–6; Tables 4 and 6

Table 4. Cross-budget Product@4 versus Euclidean@32 FID summary.

Domain/Capacity	Product@4 FID	Euclidean@32 FID	Gap (Euc32–Prod4)	Cross-Budget Passes	Product Optimum
CIFAR-100/32 M	150.13 ± 5.37	158.23 ± 8.16	8.09 ± 5.29	3/3	NFE = 8, 3/3
CIFAR-100/129 M	130.62 ± 14.05	152.98 ± 10.94	22.36 ± 19.05	3/3	NFE = 8, 3/3
Pile-driver/32 M	252.31 ± 1.84	319.02 ± 4.03	66.72 ± 2.74	3/3	NFE = 4, 3/3
Overall	-	-	-	9/9	Early Product optimum in 9/9 runs

Table 5. Target-reweighting attribution ablation on CIFAR-100 using the 32 M models.

NFE	E0 FID	E1 FID	P1 FID	P0 FID	E0−E1	P1−P0
1	262.09 ± 29.75	252.79 ± 25.19	395.50 ± 3.90	397.97 ± 13.81	9.31	−2.46
2	219.28 ± 23.60	201.62 ± 23.58	245.63 ± 12.60	250.58 ± 4.22	17.66	−4.95
4	160.47 ± 14.97	135.75 ± 11.59	158.67 ± 6.12	147.84 ± 5.08	24.72	10.83
8	150.16 ± 9.95	125.94 ± 6.72	155.52 ± 4.48	124.56 ± 1.48	24.22	30.96
16	153.35 ± 8.31	123.13 ± 7.87	166.02 ± 7.30	131.39 ± 1.94	30.22	34.63
32	155.30 ± 8.04	119.71 ± 9.16	170.03 ± 7.18	135.05 ± 2.24	35.60	34.98

Note: FID values are mean ± standard deviation across three training seeds. Positive E0−E1 and P1−P0 values indicate improvement from target reweighting. E1–P0 comparisons do not isolate geometry because representation, prior, and sampler also differ.

Table 6. Pile-domain FID and KID across integration budgets.

NFE	Product FID	Euclidean FID	FID Gap	Product KID	Euclidean KID
1	346.15 ± 11.49	357.66 ± 5.49	11.51	0.4392 ± 0.0179	0.4595 ± 0.0090
2	263.34 ± 2.25	309.78 ± 1.31	46.44	0.3338 ± 0.0037	0.4078 ± 0.0038
4	252.31 ± 1.84	308.46 ± 3.95	56.16	0.3138 ± 0.0030	0.4072 ± 0.0062
8	279.19 ± 1.14	314.76 ± 3.79	35.56	0.3520 ± 0.0015	0.4143 ± 0.0057
16	286.97 ± 0.73	317.60 ± 3.89	30.62	0.3626 ± 0.0009	0.4176 ± 0.0060
32	290.18 ± 0.55	319.02 ± 4.03	28.85	0.3670 ± 0.0008	0.4194 ± 0.0063
64	291.72 ± 0.43	319.67 ± 4.05	27.95	0.3690 ± 0.0008	0.4202 ± 0.0064
128	292.47 ± 0.41	320.04 ± 4.11	27.57	0.3701 ± 0.0008	0.4207 ± 0.0065

4. Results

4.1. Benchmark Quality–Computation Frontier

Figure 2 presents the creator-run CIFAR NFE spectra used for the primary analysis. The Product model did not outperform the Euclidean baseline at all integration budgets, particularly at NFE = 1; its advantage emerged in the low-to-moderate NFE regime rather than across the full range of integration budgets. The central cross-budget comparison was Product@4 versus Euclidean@32.

For the 32 M models, Product@4 achieved FID 150.13 ± 5.37, compared with 158.23 ± 8.16 for Euclidean@32. For the 129 M models, the corresponding values were 130.62 ± 14.05 and 152.98 ± 10.94.

The Product@4 $\leq$ Euclidean@32 FID ordering held in all six CIFAR architecture–training-seed runs.

Product FID reached its minimum at NFE = 8 in all three training seeds at both capacities, giving six of six CIFAR architecture–training-seed runs with the same early Product optimum. The creator timing columns also show that Euclidean@32 required, on average, 7.93$\times$ the Product@4 sampling time for the 32 M models and 7.98$\times$ for the 129 M models. These ratios describe generation/sampling time only, not training time or end-to-end deployment latency.

KID broadly supported the CIFAR low-NFE trend but did not provide a universal Product@4 versus Euclidean@32 crossover. Consequently, the benchmark cross-budget claim is stated specifically in terms of FID.

4.2. Target Reweighting Accounts for a Substantial Part of the Low-NFE Gain

The four-condition ablation materially changes the interpretation of the original Product–Euclidean comparison. Adding the factor-wise target-speed reweighting analog to the Euclidean model (E0$\rightarrow$E1) reduced the mean FID at every evaluated NFE. The mean paired FID reduction across the three training seeds was 9.31, 17.66, 24.72, 24.22, 30.22, and 35.60 at NFE = 1, 2, 4, 8, 16, and 32, respectively.

Most importantly, E1 at NFE = 4 reproduced the original cross-budget advantage over E0 at NFE = 32 in all three training seeds: 125.50 versus 146.10 for seed 42, 148.33 versus 158.84 for seed 123, and 133.43 versus 160.97 for seed 456. Thus, this particular low-NFE crossover can be reproduced without the Product representation under the evaluated Euclidean reweighting analog.

Target reweighting also contributed within the Product formulation. Relative to the unreweighted P1 condition, P0 reduced the mean FID by 10.83, 30.96, 34.63, and 34.98 points at NFE = 4, 8, 16, and 32, respectively. At NFE = 1 and 2, however, the mean differences were -2.46 and -4.95, respectively; the Product reweighting advantage therefore did not extend uniformly to the coarsest integration budgets.

E1 and P0 showed little separation in mean FID at NFE = 8 (125.94 versus 124.56), while E1 was lower at NFE = 4, 16, and 32. These cross-formulation differences are descriptive only because E1 and P0 still differ jointly in representation, prior, and sampler. Accordingly, the ablation identifies target-speed reweighting as a major contributor to absolute low-NFE quality but does not isolate a residual causal effect of manifold geometry.

4.3. Numerical Refinement and Generative Fidelity Are Empirically Decoupled

Figure 3 uses the same-initial-condition diagnostic evaluation rather than the creator spectra of Figure 2. Within each trained model, a fixed evaluation realization (seed 9001) is reused across NFE so that the upper FID curves and lower step-doubling discrepancies refer to the same generated realization.

In both the 32 M and 129 M Product models, FID reaches an early minimum at NFE = 8 and then worsens as the integration grid is further refined. In contrast, step-doubling LPIPS decreases monotonically across the successive refinement pairs. At the seed level, this monotone LPIPS refinement holds for all 12 CIFAR geometry–capacity–training-seed trajectories.

These results do not indicate a failure of numerical refinement. Rather, successive discrete solutions continue to approach one another even after the distribution-level quality metric has passed its optimum. This establishes an empirical separation between finite-step trajectory refinement and generative fidelity. It does not establish convergence to the exact continuous-time solution and does not identify a causal mechanism for the finite-step optimum.

4.4. Independent Checkpoint Replay Preserves the Low-NFE Ordering

The 129 M checkpoints were independently regenerated under evaluation seeds 9001, 9002, and 9003. Within each training seed, these evaluation seeds are treated as nested Monte-Carlo repeats rather than independent model replicates. Averaging the repeats within each trained model preserves Product@4 below Euclidean@32 for all three training seeds, and the Product NFE = 8 optimum is retained. Thus, the main CIFAR ordering is not specific to one historical generation realization.

4.5. Structured Construction-Domain Replication

The pile corpus contains 4000 unique synthetic images in a balanced four-material by ten-scene design, with 3200 training and 800 holdout images. Figure 4a contrasts representative 768 $\times$ 768 source images with their corresponding 32 $\times$ 32 model inputs. The blur visible in the model-input insets is therefore a consequence of resolution matching, rather than a property of the source synthetic corpus or an output of PMFM generation. A semantic-retention check gives four-way material accuracy 0.604 with macro-F1 0.558 and ten-way scene-category accuracy 0.917. As illustrated in Figure 4a, the 32 $\times$ 32 inputs preserve broad scene layout, dominant color organization, and coarse foreground/background structure, whereas fine material texture and detailed pile geometry are substantially reduced by downsampling. The higher scene-category retention relative to the material-class probe is consistent with this qualitative observation. These results therefore support retention of coarse scene-level information but not engineering-grade material identification or fine structural fidelity.

The qualitative generation sequences in Figures 5 and 6 provide a complementary view of this structured-domain behavior. Across increasing NFE, the samples develop and stabilize coarse scene-level characteristics, including broad foreground/background organization and the dominant color structure of the construction corpus. In contrast, fine pile geometry and material-specific texture remain difficult to distinguish at the evaluated 32 $\times$ 32 resolution. This qualitative pattern is consistent with the substantially higher scene-category retention after resizing (0.917) than the material-class retention indicated by the four-way probe (accuracy 0.604; macro-F1 0.558). Accordingly, these panels are interpreted as evidence of coarse scene-level generation rather than the preservation of engineering-grade material identity, object geometry, or field-relevant visual detail.

Using the same 32 M architecture and manifold operators with an exposure-matched 5120-step training horizon, the Product model achieved lower FID and KID than the Euclidean baseline in all 24 matched-NFE seed-level comparisons for each metric over NFE $\{1, 2, 4, 8, 16, 32, 64, 128\}$.

At NFE = 4, Product FID is 252.31 ± 1.84, compared with 308.46 ± 3.95 for Euclidean at the same budget. The cross-budget comparison Product@4 versus Euclidean@32 is 252.31 ± 1.84 versus 319.02 ± 4.03 and passes in all three pile training seeds.

Combined with the six CIFAR runs, Product@4 $\leq$ Euclidean@32 therefore holds in all nine evaluated domain/capacity–training-seed configurations.

4.6. The Finite-Step Optimum Is Robust in Existence but Not in Location

Extending the pile grid to NFE = 1 and 2 confirms that NFE = 4 is an interior Product optimum rather than a left-boundary artifact. Product FID at NFE = 4 was lower than that at the next-best low-NFE setting by 10.20, 7.42, and 15.47 FID points for training seeds 42, 123, and 456, respectively.

The numerical location of the Product optimum therefore shifts across domains: NFE = 8 on CIFAR and NFE = 4 on the pile corpus. Thus, the reproducible feature is the existence of an early finite-step optimum and a favorable low-NFE quality–computation frontier, not a universal integration count.

4.7. Pile-Domain Refinement and Qualitative Behavior

The pile-domain step-doubling diagnostic reproduces the benchmark pattern. LPIPS decreases monotonically in all six pile Product/Euclidean geometry–training-seed trajectories from the $1 \rightarrow 2$ comparison through $64 \rightarrow 128$, although Product FID has already reached its optimum at NFE = 4. Step refinement therefore continues after distribution-level quality begins to deteriorate.

Figures 5 and 6 separate the Product and Euclidean fixed-initial-condition progressions to improve the visibility of within-geometry refinement behavior. For the Product model (Figure 5), coarse scene-level organization becomes progressively more apparent between NFE = 1 and NFE = 4, followed by smaller visual changes at NFE = 8 and 32. Fine pile geometry and material-specific appearance remain unresolved at this resolution. The Euclidean progression (Figure 6) is shown separately under its own fixed prior realizations. These panels are intended only as qualitative illustrations: rows are held fixed across NFE within a geometry, but Product and Euclidean rows are not treated as paired cross-geometry evidence because their priors differ.

5. Discussion

5.1. The Reproducible Result Is a Low-Budget Quality Frontier

The original Product-versus-Euclidean experiments establish a reproducible low-budget ordering: Product@4 achieves lower FID than the original Euclidean@32 baseline in all nine evaluated domain/capacity–training-seed configurations across the two CIFAR capacities and the pile-domain replication. The attribution ablation, however, shows that this ordering should not be interpreted as a Product-specific or geometry-specific efficiency advantage. In the 32 M CIFAR experiment, the reweighted Euclidean condition E1 reproduces the NFE = 4 versus E0@32 crossover in all three training seeds and shows similar or lower mean FID than P0 over much of the evaluated NFE range. The robust empirical result is therefore the existence of a favorable low-budget frontier in the implemented systems, with target-speed reweighting accounting for a substantial part of the original Product-versus-baseline difference.

5.2. Numerical Refinement and Distribution-Level Quality Are Different Quantities

The same-initial-condition diagnostic shows that successive discrete solutions continue to approach one another while FID has already passed its minimum. Numerical step refinement concerns the proximity of discrete trajectories under a fixed learned field,

whereas FID measures the distribution produced by those trajectories. The experiments demonstrate that these quantities need not improve monotonically together at finite NFE.

The observed early optimum is consistent with an interaction between the trained field and the discrete sampler at finite step sizes, but the experiments do not establish a causal mechanism. In particular, they do not show that the discretization error compensates for errors in the learned continuous-time field.

5.3. The Exact Optimum Is Domain-Dependent

Product FID is minimized at NFE = 8 across all six CIFAR architecture–training-seed runs and at NFE = 4 across all three pile training seeds. The pile NFE = 1 and NFE = 2 extension confirms that NFE = 4 is an interior optimum rather than an evaluation-grid boundary. Thus, the reproducible observation is the existence of an early finite-step optimum; its numerical location remains data dependent.

The attribution ablation is not used as a same-initial-condition refinement test because its evaluation realization was matched within variant contrasts at each NFE rather than held fixed across the NFE sweep. Claims concerning the location of the finite-step optimum therefore rely on the dedicated creator and fixed-initial-condition evaluation lineages described above.

5.4. Implementation-Specific Target Construction Limits Causal Attribution

The Product results are properties of the evaluated PMFM implementation, not a causal isolation of hyperbolic curvature. Relative to the Euclidean baseline, the Product system jointly changes the state representation, prior distribution, conditional target construction, and manifold-valued sampling update. The code-level operator historically named parallel transport factor is not Levi-Civita parallel transport. After explicit destination projection, the implemented target is collinear with the geodesic tangent but is multiplied by the positive factor $2\cosh(td) - 1$, where d is the endpoint geodesic distance and t the interpolation time.

The targeted ablation resolves one part of this attribution problem. Within the Euclidean formulation, introducing the factor-wise target-speed reweighting analog improves FID at every evaluated NFE and reproduces the NFE = 4 versus E0@32 crossover in all three training seeds. Within the Product formulation, the implemented reweighting substantially improves FID from NFE ≥ 4. Target-speed reweighting is therefore a major contributor to the observed low-NFE quality gain.

The ablation does not, however, isolate the remaining Product-specific components. E1 and P0 still differ jointly in state representation, prior distribution, and sampling update. Consequently, differences between these conditions cannot be attributed specifically to hyperbolic curvature, and the present evidence does not establish a curvature-only mechanism. The Product formulation may affect the shape of the finite-step quality frontier through the coupled representation–prior–sampler system, but identifying the individual contribution of those components would require further controlled interventions.

5.5. Capacity and Domain Evidence Provide Complementary Robustness Tests

The two CIFAR capacities vary model scale while preserving the data domain. The pile study changes the image distribution while retaining the same 32 M architecture and manifold operators, with the training horizon adjusted only to preserve approximately the benchmark number of dataset passes. These two axes support a broader robustness statement than either one alone, while avoiding absolute FID comparison across domains.

5.6. Construction-Domain Interpretation and Boundaries

The pile study is a structured synthetic construction-domain replication rather than real-world field validation. The corpus retains strong scene-category information and coarse material information after 32×32 resizing, but it does not establish high-resolution material recognition or deployment readiness. Its role in the present study is to test whether the finite-step behavior persists outside CIFAR under the same model design.

5.7. Statistical and Metric Boundaries

Training seeds are the independent trained-model units. Evaluation seeds are nested Monte-Carlo repeats and are not counted as independent training replicates. FID is the primary metric for the CIFAR cross-budget claim because CIFAR KID does not provide a universal NFE = 4 crossover. The early CIFAR finite-step optimum is therefore treated as an FID-defined empirical result rather than a metric-universal optimum. On the pile corpus, FID and KID agree in all matched-NFE comparisons, and Product KID is also minimized at NFE = 4. Absolute pile FID is not compared numerically with CIFAR FID because the reference distributions and holdout sizes differ.

5.8. Limitations

The study remains limited in several respects. The pile corpus is synthetic and visually more homogeneous than field-acquired construction imagery. Architectural comparability requires 32×32 inputs. The refinement diagnostic establishes empirical Cauchy-like behavior of successive discrete solutions but does not prove convergence to an exact continuous-time solution. The mechanism behind the finite-step optimum remains unresolved. The new target-reweighting ablation shows that a substantial part of the original low-NFE quality gain is reproducible without the Product representation, while representation, prior, and sampler remain jointly changed in the residual Product-versus-Euclidean comparison. Future work should therefore isolate these remaining components, examine higher-order manifold integrators, and evaluate higher-resolution and field-acquired data to further characterize the finite-step behavior.

The pile-domain experiment remains a synthetic-domain replication. Although its balanced material–scene design provides a structured distribution distinct from CIFAR-100, neither the semantic-retention probes nor the generated samples establish transfer to field-acquired pile-driver imagery. In particular, synthetic-to-field differences in texture, illumination, background clutter, sensor characteristics, and fine material appearance were not evaluated. The pile-domain results therefore support replication of the observed finite-step behavior on a structured engineering-style image distribution, not field generalization or deployment readiness.

The present experiments do not establish that the reported finite-step ordering is invariant to image resolution. The 32×32 input was retained to preserve architectural comparability with the CIFAR experiments, including the same patch and Product-factor structure; evaluating higher resolutions would change the token count and computational regime and is therefore treated as a separate sensitivity question rather than assumed to preserve the present conclusions. Within this fixed-resolution setting, the retention analyses indicate that coarse scene information remains measurable after downsampling, whereas fine material texture and detailed pile geometry are substantially reduced.

6. Conclusions

This study examines the relationship between numerical integration budget and generative fidelity in an explicitly documented Product-manifold generative flow implementation. Across two CIFAR-100 model capacities and three training seeds per capacity,

the original Product@4 condition achieves lower FID than the original Euclidean@32 baseline in all six architecture–training-seed runs. The same ordering is reproduced in all three pile-domain training seeds, giving nine of nine evaluated domain/capacity–training-seed configurations. A targeted 32 M attribution ablation, however, shows that this crossover is not unique to the Product formulation: a factor-wise Euclidean analog of the Product target-speed reweighting reproduces the NFE = 4 versus Euclidean@32 advantage in all three training seeds.

The central numerical finding is that finer step refinement does not imply monotonically better distribution-level quality. Product FID reaches an early optimum at NFE = 8 on CIFAR and NFE = 4 on the pile corpus, while same-initial-condition step-doubling discrepancies continue to decrease. The existence of the early optimum therefore generalizes across capacities and domains, whereas its exact location is domain-dependent.

The ablation identifies target-speed reweighting as a major contributor to the absolute low-NFE quality gain: it improves the Euclidean baseline throughout the evaluated NFE range and also benefits the Product formulation from NFE ≥ 4. We therefore do not attribute the original Product-versus-Euclidean crossover to manifold geometry or curvature alone. Representation, prior, and sampling update remain coupled in the residual cross-formulation comparison and are not causally separated by the present experiments.

The separate same-initial-condition diagnostics nevertheless retain the central numerical observation of this study: distribution-level Product FID reaches an early finite-step optimum while successive discretizations continue to approach one another. The most defensible conclusion is therefore twofold: target-speed reweighting accounts for a substantial part of the observed low-NFE quality advantage, while the evaluated Product formulation exhibits a reproducible finite-step quality optimum whose location varies across the studied domains.

These conclusions are supported by five distinct evaluation lineages, summarized in Table 3, rather than by pooling all evaluations as equivalent replicates: the creator CIFAR spectra establish the primary NFE ordering, checkpoint replay evaluates sensitivity to generation-level Monte-Carlo variation, the fixed-initial-condition diagnostic examines numerical step refinement, independently trained pile-domain models provide structured-domain replication, and the targeted E0/E1/P1/P0 ablation evaluates the contribution of target-speed reweighting.

Author Contributions: Conceptualization, H.J., Y.C. and S.C.; methodology, H.J. and Y.C.; software, H.J.; validation, H.J. and Y.C.; formal analysis, H.J.; investigation, H.J. and Y.C.; resources, Y.C. and S.C.; data curation, H.J.; visualization, H.J.; writing—original draft preparation, H.J.; writing—review and editing, H.J., Y.C. and S.C.; supervision, S.C.; project administration, S.C.; funding acquisition, S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Machinery Industry Technology Development Program (No. 00421895) funded by the Ministry of Trade, Industry & Energy (MOTIE, Korea) and the Korea Evaluation Institute of Industrial Technology (KEIT).

Data Availability Statement: The CIFAR-100 dataset used in this study is publicly available from its original source. The author-generated synthetic pile-driver image corpus and the numerical data supporting the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: During the preparation of this manuscript, the authors used generative AI tools to assist with manuscript organization, language refinement, analysis-code drafting, and statistical review. All computational analyses, technical decisions, interpretation of results, and conclusions were independently performed and verified by the authors, who take full responsibility for the integrity and content of the published work.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FM	Flow Matching
PMFM	Product Manifold Flow Matching
NFE	Number of Function Evaluations
FID	Fréchet Inception Distance
KID	Kernel Inception Distance
LPIPS	Learned Perceptual Image Patch Similarity
RMSE	Root Mean Square Error
ODE	Ordinary Differential Equation
DiT	Diffusion Transformer

References

1. Lipman, Y.; Chen, R.T.Q.; Ben-Hamu, H.; Nickel, M.; Le, M. Flow Matching for Generative Modeling. In Proceedings of the International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023.
2. Hairer, E.; Nørsett, S.P.; Wanner, G. *Solving Ordinary Differential Equations I: Nonstiff Problems*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 1993. [CrossRef]
3. Chen, R.T.Q.; Lipman, Y. Flow Matching on General Geometries. In Proceedings of the International Conference on Learning Representations (ICLR), Vienna, Austria, 7–11 May 2024.
4. Liu, X.; Gong, C.; Liu, Q. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In Proceedings of the International Conference on Learning Representations (ICLR), Kigali, Rwanda, 1–5 May 2023.
5. Pooladian, A.-A.; Ben-Hamu, H.; Domingo-Enrich, C.; Amos, B.; Lipman, Y.; Chen, R.T.Q. Multisample Flow Matching: Straightening Flows with Minibatch Couplings. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; PMLR 202, pp. 28100–28127.
6. Tong, A.; Fatras, K.; Malkin, N.; Huguet, G.; Zhang, Y.; Rector-Brooks, J.; Wolf, G.; Bengio, Y. Improving and Generalizing Flow-Based Generative Models with Minibatch Optimal Transport. *arXiv* **2024**, arXiv:2302.00482.
7. Song, Y.; Dhariwal, P.; Chen, M.; Sutskever, I. Consistency Models. In Proceedings of the 40th International Conference on Machine Learning, Honolulu, HI, USA, 23–29 July 2023; PMLR 202.
8. Ganea, O.-E.; Bécigneul, G.; Hofmann, T. Hyperbolic Neural Networks. In *Advances in Neural Information Processing Systems 31*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018.
9. Mathieu, E.; Le Lan, C.; Maddison, C.J.; Tomioka, R.; Teh, Y.W. Continuous Hierarchical Representations with Poincaré Variational Auto-Encoders. *Adv. Neural Inf. Process. Syst.* **2019**, *32*, 12542–12552.
10. De Bortoli, V.; Mathieu, E.; Hutchinson, M.; Thornton, J.; Teh, Y.W.; Doucet, A. Riemannian Score-Based Generative Modelling. In *Advances in Neural Information Processing Systems*; Curran Associates, Inc.: Red Hook, NY, USA, 2022; Volume 35, pp. 2406–2422.
11. Xu, L.; Liu, H.; Xiao, B.; Luo, X.; Veeramani, D.; Zhu, Z. A Systematic Review and Evaluation of Synthetic Simulated Data Generation Strategies for Deep Learning Applications in Construction. *Adv. Eng. Inform.* **2024**, *62*, 102699. [CrossRef]
12. Lee, J.G.; Hwang, J.; Chi, S.; Seo, J. Synthetic Image Dataset Development for Vision-Based Construction Equipment Detection. *J. Comput. Civ. Eng.* **2022**, *36*, 04022020. [CrossRef]
13. Tohidifar, A.; Kim, D.; Lee, S.H. Make It Till You Fake It: Construction-Centric Computational Framework for Simultaneous Image Synthetization and Multimodal Labeling. *Autom. Constr.* **2024**, *167*, 105696. [CrossRef]
14. Han, S.; Park, W.; Jeong, K.; Hong, T.; Koo, C. Utilizing Synthetic Images to Enhance the Automated Recognition of Small-Sized Construction Tools. *Autom. Constr.* **2024**, *163*, 105415. [CrossRef]
15. Song, S.; Hong, J.; Jeoung, J.; Ahn, J.; Hong, T. Data-Centric Enhancement of Site-Specific Automated Construction Equipment Detection in Wide-Angle Site Images. *Autom. Constr.* **2025**, *179*, 106483. [CrossRef]
16. Krizhevsky, A.; Hinton, G. *Learning Multiple Layers of Features from Tiny Images*; Technical Report; University of Toronto: Toronto, ON, Canada, 2009.
17. Peebles, W.; Xie, S. Scalable Diffusion Models with Transformers. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, 1–6 October 2023; pp. 4195–4205. [CrossRef]
18. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the International Conference on Learning Representations (ICLR), San Diego, CA, USA, 7–9 May 2015.

19. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 6626–6637.
20. Bińkowski, M.; Sutherland, D.J.; Arbel, M.; Gretton, A. Demystifying MMD GANs. In Proceedings of the International Conference on Learning Representations (ICLR), Vancouver, BC, Canada, 30 April–3 May 2018.
21. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 586–595.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.