AdvFM: Lookahead Flow-Matching Velocity-Field Attacks for Imperceptible and Transferable Adversarial Examples

Supplementary Material

8. Detailed Theoretical Analysis

8.1. Detailed Lookahead Attack Formulation

The lookahead variant introduced in Sec. 3.2.2 couples the gradient at the current reconstruction with a short forward rollout, enabling each update to anticipate how the velocity field will transport the perturbation. For clarity, we summarize the linearization used in our analysis; full proofs are provided in Sec. 8.2. Our derivation proceeds in four steps. (i) We formulate the two-point objective as a mixture between the loss at x_1^t and the loss at its rolled-out counterpart $x_1^{t+\Delta t}$; (ii) The gradient of this objective becomes a weighted sum of two standard gradients; (iii) The noise components arising from these surrogate gradients similarly combine as a weighted average; (iv) This averaging reduces variance and, by the variance–alignment principle, yields improved cosine alignment with both the target gradient and the robust gradient. Next, we will conduct a detailed analysis of each step.

8.1.1. Two-point rollout and linearization

At time t, the reconstruction of x_t at $t = 1$ is

$$x_1^t = R_t(x_t) = x_t + (1-t)\, v_\theta(x_t, t). \tag{32}$$

AdvFM transports this reconstruction backward through the flow. Without any attack, the next noisy state and its reconstruction are

$$x_{t+\Delta t}^0 = x_t + \Delta t\, v_\theta(x_t, t), \tag{33}$$

and

$$\begin{aligned} x_1^{t+\Delta t,0} &= R_{t+\Delta t}(x_{t+\Delta t}^0) \\ &= x_{t+\Delta t}^0 + (1-t-\Delta t)\, v_\theta(x_{t+\Delta t}^0, t+\Delta t). \end{aligned} \tag{34}$$

When a perturbation δ is injected at x_1^t, the attacked velocity is given by (8) and the next noisy state becomes

$$x_{t+\Delta t}(\delta) = x_t + \Delta t\, \hat{v}_t(\delta) = x_{t+\Delta t}^0 + \frac{\Delta t}{1-t}\,\delta. \tag{35}$$

For small Δt, the reconstruction map $R_{t+\Delta t}$ is approximately linear in a neighborhood of $x_{t+\Delta t}^0$, so

$$x_1^{t+\Delta t}(\delta) = R_{t+\Delta t}(x_{t+\Delta t}(\delta)) \approx x_1^{t+\Delta t,0} + A_t\,\delta, \tag{36}$$

where $A_t := \frac{\Delta t}{1-t}\, I$. This first-order view links the perturbation we optimize at t to the future reconstruction that the classifier will actually see.

8.1.2. Gradient structure of the two-point loss

The lookahead loss in (12) averages the classification loss at x_1^t and at its rolled-out counterpart $x_1^{t+\Delta t}(\delta)$. The gradient of this objective around $\delta = 0$ is a weighted combination of the two reconstructions.

Lemma 5 (Gradient of the two-point lookahead loss)**.** *Under the linear approximation* (36)*,*

$$\begin{aligned} \nabla_\delta \mathcal{L}_f^{LA}(0;t) &= w\, \mathcal{G}_f(x_1^t) + (1-w)\, A_t^\top \mathcal{G}_f\big(x_1^{t+\Delta t,0}\big) \\ &= w\, \mathcal{G}_f(x_1^t) + (1-w)\, \frac{\Delta t}{1-t}\, \mathcal{G}_f\big(x_1^{t+\Delta t,0}\big). \end{aligned} \tag{37}$$

Thus the lookahead direction trades off a 'now' gradient and a 'rolled-out' gradient scaled by the flow step; w interpolates between a greedy update (w=1) and one that anticipates transport along the velocity field. This is the core design choice: mixing two nearby reconstructions to mirror what the classifier will actually see.

8.1.3. Noise decomposition

Next we make explicit how gradient noise behaves when we mix the two reconstructions. Let $\mathcal{G}_g(x)$ be the target gradient, and suppose the surrogate gradients can be written as $\mathcal{G}_f(x_1^t) = \mathcal{G}_g(x_1^t) + \eta_1$ and $\mathcal{G}_f(x_1^{t+\Delta t,0}) = \mathcal{G}_g(x_1^{t+\Delta t,0}) + \eta_2$, where η_1, η_2 are zero-mean, independent, isotropic noise terms with covariance $\sigma^2 I$. This is the same noise model used in Sec. 4 to relate variance to alignment.

Lemma 6 (Noise structure of the lookahead direction)**.** *Define the two-point target gradient*

$$h_t := w\, \mathcal{G}_g(x_1^t) + (1-w)\, \frac{\Delta t}{1-t}\, \mathcal{G}_g\big(x_1^{t+\Delta t,0}\big). \tag{38}$$

Then the baseline and lookahead directions can be written as

$$u_t^{VF} = \mathcal{G}_g(x_1^t) + \eta_1, \tag{39}$$

$$u_t^{LA} = h_t + \zeta_t, \tag{40}$$

where

$$\zeta_t = w\,\eta_1 + (1-w)\, \frac{\Delta t}{1-t}\, \eta_2, \tag{41}$$

and

$$\mathrm{Var}[\zeta_t] = \sigma^2 \Big(w^2 + (1-w)^2 \frac{\Delta t^2}{(1-t)^2} \Big). \tag{42}$$

8.1.4. Variance reduction and alignment

Finally, we quantify how mixing two noisy gradients affects variance and alignment. The next lemma shows the variance shrinks under the lookahead weighting, and the corollary connects this to cosine alignment via Lemma 4.

Lemma 7 (Variance reduction for the two-point objective)**.** *If $w \in (0,1)$ and the step schedule satisfies $\Delta t < 1 - t$, then*

$$\mathrm{Var}[\zeta_t] < \sigma^2 = \mathrm{Var}[\eta_1], \tag{43}$$

so the lookahead direction has lower variance than the single-point update with respect to either h_t or $\mathcal{G}_g(x_1^t)$.

Corollary 1 (Alignment with robust or two-point gradients)**.** *Lemmas 5, 6, and 7 together imply that the lookahead direction is a lower-variance estimator of the two-point target gradient. Combined with the variance-to-alignment relation in Lemma 4, it satisfies*

$$\mathbb{E}\big[\cos\angle(u_t^{LA}, h_t)\big] > \mathbb{E}\big[\cos\angle(u_t^{VF}, h_t)\big], \tag{44}$$

and the same ordering holds when replacing h_t with $\mathcal{G}_{rob}$ under the robustness alignment assumption used in Theorem 2.

Together, these results explain why lookahead improves the black-box performance seen in Sec. 5.4: it predicts how the perturbation will propagate, mixes gradients from two correlated reconstructions, and thereby reduces noise without sacrificing the tangential structure emphasized by flow matching.

8.2. Theoretical Proofs

We provide proofs of all statements from Sec. 4 and the auxiliary lemmas above.

8.2.1. Proof of Lemma 1

From (11) and (4) we have

$$\begin{aligned}\Delta\mathcal{L}_g^{\mathrm{FM}} &\approx \frac{\Delta t}{1-t}\,\nabla_x\mathcal{L}_g(x_t,y)^\top\delta,\\ \Delta\mathcal{L}_g^{\mathrm{Diff}} &\approx \sqrt{\bar\alpha_t}\,\nabla_x\mathcal{L}_g(x_t,y)^\top\delta,\end{aligned} \tag{45}$$

so

$$\frac{\Delta\mathcal{L}_g^{\mathrm{FM}}}{\Delta\mathcal{L}_g^{\mathrm{Diff}}} \approx \frac{\Delta t}{(1-t)\sqrt{\bar\alpha_t}}. \tag{46}$$

Along the reverse trajectory, $t \to 1$ and $1-t$ becomes small, while Δt and $\bar\alpha_t \in (0,1)$ are fixed by the schedule. Thus $\Delta t/(1-t) > 1$ and $\sqrt{\bar\alpha_t} < 1$, which yields (17).

8.2.2. Proof of Lemma 2

By the reparameterization $x_t = tx + (1-t)\epsilon$ and the chain rule,

$$\begin{aligned}\nabla_x\tilde{\mathcal{L}}_f(x) &= \mathbb{E}_\epsilon\left[\nabla_{x_t}\mathcal{L}_f(x_t,y)^\top\frac{\partial x_t}{\partial x}\right]\\ &= \mathbb{E}_\epsilon\left[\nabla_{x_t}\mathcal{L}_f(x_t,y)^\top tI\right]\\ &= t\,\mathbb{E}_\epsilon\left[\nabla_{x_t}\mathcal{L}_f(x_t,y)\right].\end{aligned} \tag{47}$$

8.2.3. Proof of Lemma 3

Let $Z = \nabla_x\mathcal{L}_f(x,y)$ and $Z_t = \nabla_{x_t}\mathcal{L}_f(x_t,y)$ with x_t as above. Gaussian smoothing averages gradients, so

$$\mathrm{Var}[Z] \;\geq\; \mathrm{Var}[Z_t], \tag{48}$$

as in standard results on smoothed objectives. By Lemma 2, our velocity-field attack via flow matching approximates the smoothed gradient $\nabla_x\tilde{\mathcal{L}}_f(x)$, which is an average of $\nabla_{x_t}\mathcal{L}_f(x_t,y)$; averaging reduces variance, giving

$$\mathrm{Var}\big[\mathcal{G}_f^{\mathrm{FM}}\big] \leq \mathrm{Var}\big[\nabla_{x_t}\mathcal{L}_f(x_t,y)\big]. \tag{49}$$

For the diffusion-based attack, the gradient is taken at x_0', which can be decomposed as

$$x_0' = \tilde{x}_0 - c_t\xi_t, \quad c_t = \frac{\sqrt{1-\bar\alpha_t}}{\sqrt{\bar\alpha_t}}, \tag{50}$$

where $\tilde{x}_0$ is the ideal reconstruction and ξ_t is the noise estimation error. A first-order expansion yields

$$\mathcal{G}_f^{\mathrm{Diff}} \approx \nabla_x\mathcal{L}_f(\tilde{x}_0,y) - H_f(\tilde{x}_0)\,c_t\xi_t, \tag{51}$$

with H_f the Hessian. Writing $A = \nabla_x\mathcal{L}_f(\tilde{x}_0,y)$ and $B = -H_f(\tilde{x}_0)\,c_t\xi_t$, we have

$$\mathrm{Var}\big[\mathcal{G}_f^{\mathrm{Diff}}\big] = \mathrm{Var}[A{+}B] \geq \mathrm{Var}[A] \approx \mathrm{Var}\big[\nabla_{x_t}\mathcal{L}_f(x_t,y)\big], \tag{52}$$

where the last approximation uses the local correspondence between $\tilde{x}_0$ and x_t. Combining these inequalities proves the claim.

8.2.4. Proof of Lemma 4

Write $u = \mathcal{G}_g(x)$ and $\hat{u} = u + \eta$. For isotropic zero-mean η with covariance $\sigma^2 I_d$, the numerator of the cosine is $\langle\hat{u},u\rangle = \|u\|^2 + \langle\eta,u\rangle$, whose expectation equals $\|u\|^2$. The denominator involves $\|\hat{u}\|^2 = \|u\|^2 + 2\langle u,\eta\rangle + \|\eta\|^2$, and $\mathbb{E}[\|\eta\|^2] = \sigma^2 d$. Approximating $\mathbb{E}[1/\|\hat{u}\|]$ by $1/\sqrt{\mathbb{E}[\|\hat{u}\|^2]}$ yields the stated expression, which is monotonically decreasing in σ^2.

8.2.5. Proof of Theorem 1

Lemma 3 gives $\mathrm{Var}[\mathcal{G}_f^{\mathrm{FM}}] \leq \mathrm{Var}[\mathcal{G}_f^{\mathrm{Diff}}]$. Applying Lemma 4 to both directions with the same reference $\mathcal{G}_g$ shows that the expected cosine similarity decreases as variance grows, yielding the alignment ordering claimed in Theorem 1. Combining this with Lemma 1 establishes the transferability claim.

8.2.6. Proof of Theorem 2

Suppose $\Pi_{U(x)}\mathcal{G}_{\mathrm{rob}}(x) \neq 0$. Because FM smoothing averages over a noise bridge rather than a single point, its mean gradient inside $U(x)$ is more aligned with $\Pi_{U(x)}\mathcal{G}_{\mathrm{rob}}(x)$ than the natural gradient is, with some margin $\delta > 0$. Lemma 3 provides a variance ordering between the two

Table 3. Black-box ASR (%) on CIFAR-10. Best results per column are in **bold**, second-best are underlined.

ASR (%)	Src.\Tgt.	CNNs				Transformers				Avg.
		VGG16	RN34	DN121	EN-B5	ViT-B16	Swin-S	DeiT-S	VisF-S	
AdvGAN	VGG16	75.31*	**65.59**	42.10	41.17	23.65	33.00	30.20	31.57	38.18
AdvDiffuser		80.02*	61.16	54.46	40.94	28.18	30.00	35.51	32.22	40.35
DiffPGD		77.85*	50.51	53.77	35.86	30.03	28.28	33.13	34.98	38.08
AdvAD		73.86*	45.00	42.30	**53.30**	41.20	40.40	48.20	**64.50**	47.84
DiffAttack		83.17*	54.10	**61.59**	41.56	54.49	52.19	47.89	50.58	51.77
AdvFM (ours)		**86.10***	**57.71**	60.52	51.41	**59.58**	**64.10**	**61.50**	63.42	**59.75**
AdvGAN	RN34	46.23	84.37*	60.53	50.00	26.88	34.00	33.33	41.05	41.72
AdvDiffuser		23.10	72.13*	30.12	19.50	25.61	20.08	21.10	29.92	24.20
DiffPGD		35.60	95.30*	45.69	36.05	30.32	37.10	23.75	36.31	34.97
AdvAD		47.95	97.30*	48.45	53.33	32.06	32.02	34.41	31.15	39.91
DiffAttack		43.50	95.90*	55.79	46.25	43.37	42.19	43.94	33.94	44.14
AdvFM (ours)		**74.63**	**99.40***	**82.89**	**61.12**	**44.78**	**44.56**	**45.13**	**49.49**	**57.51**
AdvGAN	ViT-B16	33.33	28.12	34.21	36.76	78.06*	42.00	43.75	32.63	35.83
AdvDiffuser		24.84	20.17	31.98	27.08	80.00*	45.35	47.47	50.95	35.41
DiffPGD		20.11	25.66	30.34	28.64	79.92*	40.05	35.55	30.68	30.15
AdvAD		30.23	31.47	42.82	44.00	98.97*	30.10	40.66	35.83	36.44
DiffAttack		31.14	40.61	42.27	39.06	98.87*	49.23	54.47	**55.00**	44.54
AdvFM (ours)		**43.42**	**44.17**	**47.52**	**45.99**	**99.10***	**59.92**	**75.32**	52.54	**52.70**
AdvGAN	Swin-S	30.78	34.37	31.15	32.70	33.33	78.00*	34.37	30.84	32.51
AdvDiffuser		31.13	30.11	35.48	33.56	41.46	85.48*	45.11	40.42	36.75
DiffPGD		29.92	32.11	27.89	31.66	34.24	84.27*	41.34	39.35	33.79
AdvAD		**56.80**	**49.40**	41.41	32.67	35.15	98.60*	49.64	47.53	44.66
DiffAttack		13.11	29.62	31.88	15.62	30.33	86.04*	28.94	55.29	29.26
AdvFM (ours)		54.62	42.30	**45.08**	**43.30**	**48.69**	**98.70***	**51.98**	**61.54**	**49.64**

attack directions, and Lemma 4 implies that the expected cosine similarity with any fixed reference vector (here $\mathcal{G}_{\text{rob}}(x)$) is a strictly decreasing function of the noise variance. Applying Lemma 4 to the two noise models yields the existence of κ_{FM} and κ_{Diff}.

8.2.7. Proof of Theorem 3

For small perturbations η, purification operators behave approximately like anisotropic projectors: tangential components are preserved more than off-manifold components. Formally, there exist constants $\lambda_U > \lambda_\perp \geq 0$, a constant $C > 0$, and a radius $\rho > 0$ such that for all $\|\eta\| \leq \rho$,

$$\|P(x+\eta) - P(x)\|^2 \;\geq\; \lambda_U \|\eta_U\|^2 + \lambda_\perp \|\eta_\perp\|^2 - C\|\eta\|^3. \tag{53}$$

We also consider the regime where both attacks have comparable total energy and FM spends at least $\epsilon > 0$ more fraction in the tangential subspace:

$$r_{\text{FM}} \;\geq\; r_{\text{Diff}} + \epsilon, \tag{54}$$

and

$$0 < c_1 \;\leq\; \frac{\sigma_{\text{FM}}^2}{\sigma_{\text{Diff}}^2} \;\leq\; c_2 < \infty \tag{55}$$

for some constants c_1, c_2.

Applying (53) to the terminal perturbation η^{method} and writing $\Delta P := P(x+\eta^{\text{method}}) - P(x)$, we have, whenever $\|\eta^{\text{method}}\| \leq \rho$,

$$\|\Delta P\|^2 \;\geq\; \lambda_U \|\eta_U^{\text{method}}\|^2 + \lambda_\perp \|\eta_\perp^{\text{method}}\|^2 - C\|\eta^{\text{method}}\|^3. \tag{56}$$

Taking expectations and using the definition of r_{method}, we obtain the stated result.

8.2.8. Proof of Lemma 5

Differentiating (12) with respect to δ and applying the chain rule yields

$$\begin{aligned}\nabla_\delta \mathcal{L}_f^{\text{LA}}(\delta; t) =& w\, \nabla_x \mathcal{L}_f(x_1^t + \delta, y) \\ &+ (1-w)\, \nabla_x \mathcal{L}_f\big(x_1^{t+\Delta t}(\delta), y\big) \frac{\partial x_1^{t+\Delta t}(\delta)}{\partial \delta}.\end{aligned} \tag{57}$$

Evaluating at $\delta = 0$ and substituting the linearization (36) gives (37).

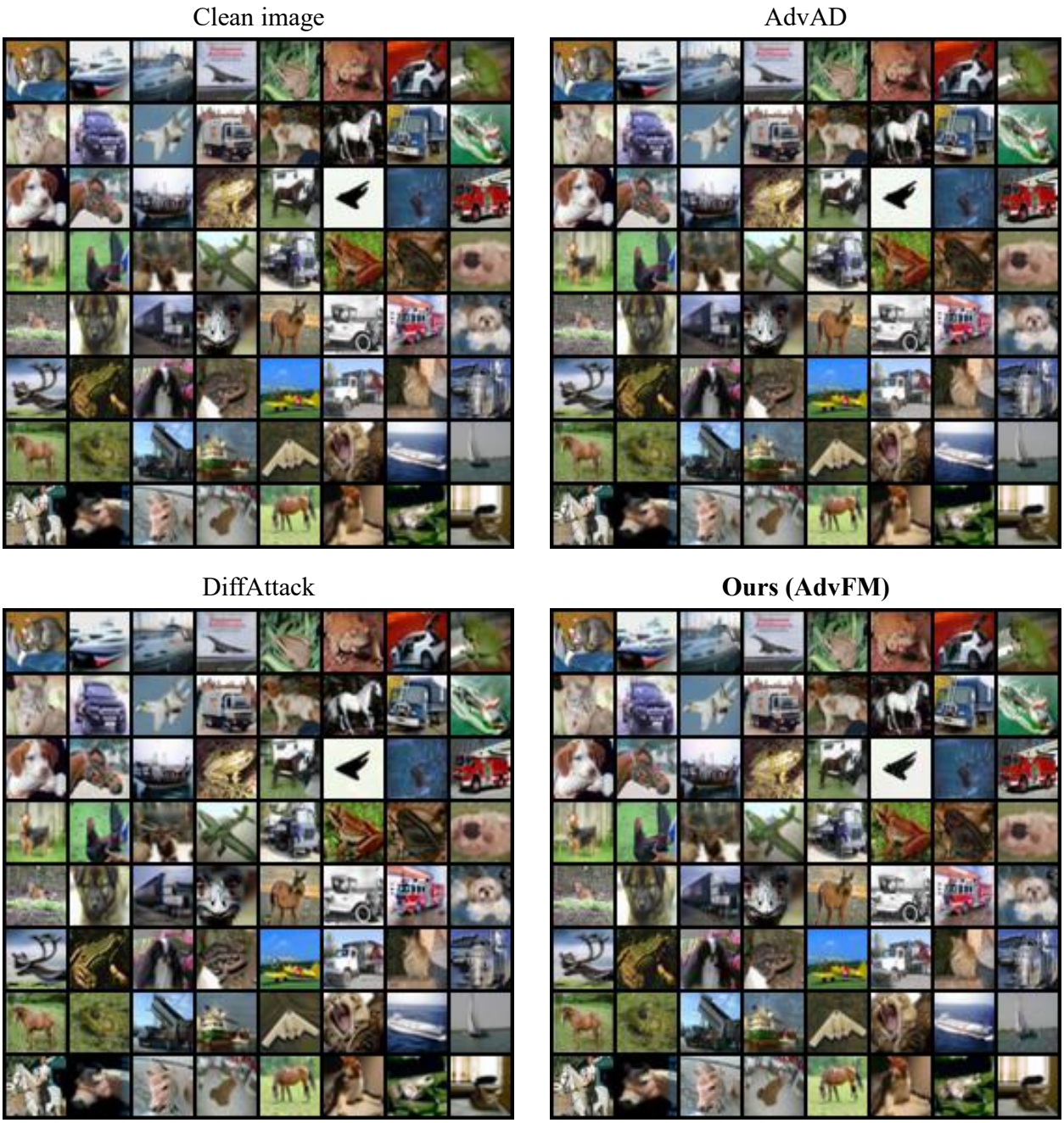

Figure 3. Qualitative comparison of adversarial examples on CIFAR-10 under ResNet34.

8.2.9. Proof of Lemma 6

By assumption, $\mathcal{G}_f(x_1^t) = \mathcal{G}_g(x_1^t) + \eta_1$ and $\mathcal{G}_f(x_1^{t+\Delta t,0}) = \mathcal{G}_g(x_1^{t+\Delta t,0}) + \eta_2$ with independent zero-mean isotropic noise. Plugging these decompositions into Lemma 5 and collecting deterministic terms yields h_t in (38) and the noise equation (41). Independence and isotropy imply the stated variance formula.

8.2.10. Proof of Lemma 7

From Lemma 6,

$$\mathrm{Var}[\zeta_t] = \sigma^2\Big(w^2 + (1-w)^2\frac{\Delta t^2}{(1-t)^2}\Big). \tag{58}$$

When $\Delta t < 1-t$, the factor $(\Delta t/(1-t))^2 < 1$, and for any $w \in (0,1)$ we have

$$w^2+(1-w)^2\frac{\Delta t^2}{(1-t)^2} < w^2+(1-w)^2 = 1-2w(1-w) < 1, \tag{59}$$

so $\mathrm{Var}[\zeta_t] < \sigma^2 = \mathrm{Var}[\eta_1]$.

8.2.11. Proof of Corollary 1

Lemma 7 shows that the noise variance in u_t^{LA} is strictly smaller than that of u_t^{VF}. Applying Lemma 4 with reference vector h_t yields the cosine ordering. When $\mathcal{G}_{\mathrm{rob}}$ replaces h_t and is positively aligned with it (as assumed in Theorem 2), the same variance ordering implies the same cosine ordering.

8.2.12. Proof of Theorem 4

Lemma 5 expresses u_t^{LA} as an unbiased estimator of the two-point gradient h_t, while u_t^{VF} estimates only the first term of h_t. Lemma 7 shows that the variance of u_t^{LA} around h_t is smaller than the variance of u_t^{VF} around either h_t or $\mathcal{G}_g(x_1^t)$. Applying Lemma 4 with reference h_t yields the expected cosine ordering claimed in Theorem 4.

8.2.13. Proof of Theorem 5

Applying the directional purification bound (53) to η^{LA} and η^{VF} (as in Theorem 3), if lookahead allocates at least as much energy to $U(x)$ as the baseline (i.e., $r_{\mathrm{LA}} \geq r_{\mathrm{VF}}$ and the total energies are comparable, analogously to (54) and (55)), the lower bound (53) yields

$$\mathbb{E}\|P(x+\eta^{\mathrm{LA}})-P(x)\|^2 \geq \mathbb{E}\|P(x+\eta^{\mathrm{VF}})-P(x)\|^2. \tag{60}$$

The alignment claim follows from the same variance argument as in Corollary 1, now with reference h_t.

9. Experiments on Other Datasets

9.1. Experiments on CIFAR-10

9.1.1. Experimental Setup

We further evaluate AdvFM on CIFAR-10 to verify that our conclusions generalize beyond ImageNet. We consider eight classifiers: four CNNs (VGG16, ResNet34, DenseNet121, EfficientNet-B5) and four vision Transformers (ViT-B/16, Swin-S, DeiT-S, VisFormer-S), implemented using standard `torchvision` and `timm` backbones. Each model is trained on CIFAR-10 and is used both as a surrogate (source) model and as a black-box target. We use the pre-trained torchcfm [32] as the backbone for flow matching. For black-box transfer, we craft UAEs from a given source model and report attack success rate (ASR) on all remaining targets. For perceptual quality, we focus on ResNet34 and measure LPIPS, PSNR, SSIM and FID between clean and adversarial images. For robustness against adversarial training, we attack four adversarially trained ResNet18 models (Madry, IAT, OA-AT, DeepMind) in the white-box setting and report ASR. All CIFAR-10 results are computed on the full test set.

Table 4. Perceptual quality of adversarial examples on ResNet34. Lower LPIPS/FID and higher PSNR/SSIM indicate better visual quality.

Method	LPIPS↓	PSNR↑	SSIM↑	FID↓
AdvGAN	0.0208	27.74	0.929	51.62
AdvDiffuser	0.0179	29.58	0.933	49.09
DiffPGD	0.0084	31.86	0.957	45.20
AdvAD	**0.0013**	<u>35.73</u>	**0.994**	<u>39.79</u>
DiffAttack	0.0059	28.67	0.942	40.08
AdvFM (ours)	<u>0.0019</u>	**37.58**	<u>0.990</u>	**38.57**

Table 5. ASR (%) against adversarially trained ResNet18 models on CIFAR-10. Best results per column are in **bold**, second-best are <u>underlined</u>.

Method	Adversarially trained ResNet18			
	Madry	IAT	OA-AT	DeepMind
AdvGAN	49.40	58.58	42.33	37.77
AdvDiffuser	60.12	58.63	41.56	35.88
DiffPGD	54.98	59.95	40.79	38.64
AdvAD	51.69	<u>60.78</u>	28.74	26.60
DiffAttack	<u>61.72</u>	46.08	<u>50.44</u>	<u>60.19</u>
AdvFM (ours)	**98.88**	**98.65**	**90.23**	**85.45**

9.1.2. CIFAR-10 Black-box Transfer Experiments

Tab. 3 shows that AdvFM consistently achieves the highest average ASR for every source model on CIFAR-10. When attacking a CNN surrogate (VGG16 or ResNet34), AdvFM not only keeps strong white-box performance (e.g., 86.10% and 99.40% on the source itself), but also substantially boosts transfer to both CNN and transformer targets. For instance, with VGG16 as the source, AdvFM improves the average ASR from 51.77% (DiffAttack) to 59.75%, with clear gains on transformer targets such as Swin-S (64.10% vs. 52.19%) and DeiT-S (61.50% vs. 47.89%). A similar trend holds for ResNet34, where AdvFM lifts the cross-architecture transfer to ViT-B16, Swin-S, DeiT-S and VisFormer-S, while also achieving the best or second-best results on all CNN targets.

When using transformer surrogates (ViT-B16 or Swin-S), AdvFM remains competitive on CNN targets and yields particularly large gains on other transformers. For example, with ViT-B16 as source, AdvFM reaches 99.10% ASR on ViT-B16, and 75.32% on DeiT-S, substantially higher than all baselines. With Swin-S as source, AdvFM attains the best average ASR (49.64%) and the strongest transfer to VisFormer-S (61.54%), while remaining close to the best

Table 6. Black-box ASR (%) on CelebA. Best results per column (within each source model block) are in **bold**, second-best are underlined.

ASR (%)	Src.\Tgt.	CNNs				Transformers				Avg.
		VGG16	RN34	DN121	EN-B5	ViT-B16	Swin-S	DeiT-S	VisF-S	
AdvGAN	VGG16	89.50*	50.00	48.33	49.35	39.91	29.39	25.12	27.36	38.49
AdvDiffuser		81.53*	**59.38**	54.66	42.53	33.44	32.79	31.40	31.95	40.88
DiffPGD		80.15*	49.96	55.01	37.45	32.01	30.11	35.35	34.98	39.27
AdvAD		**98.71***	51.52	**72.86**	41.54	47.27	40.64	39.15	40.17	47.59
DiffAttack		85.11*	58.58	58.50	44.44	58.79	54.40	42.40	48.91	52.29
AdvFM (ours)		96.10*	45.31	54.37	**52.50**	**70.63**	**66.88**	**67.81**	**68.75**	**60.89**
AdvGAN	RN34	51.67	96.00*	51.21	45.67	33.05	31.00	27.55	33.77	39.13
AdvDiffuser		32.41	80.98*	37.05	28.70	33.14	40.43	30.09	32.56	33.48
DiffPGD		34.44	94.46*	44.88	33.13	35.01	38.80	38.46	34.31	37.00
AdvAD		69.09	97.22*	40.27	58.06	46.55	50.00	36.92	33.28	47.74
DiffAttack		57.67	93.93*	47.90	59.59	46.00	41.80	46.68	43.90	49.08
AdvFM (ours)		**70.63**	**99.69***	**75.63**	**67.19**	**76.25**	**73.44**	**75.94**	**75.94**	**73.57**
AdvGAN	ViT-B16	29.80	26.67	28.26	31.43	94.20*	44.07	49.25	51.83	37.33
AdvDiffuser		34.12	30.71	35.94	31.12	84.20*	45.55	46.66	50.95	39.29
DiffPGD		31.00	35.66	43.03	36.80	79.92*	39.99	43.18	33.78	37.63
AdvAD		45.45	44.00	41.79	35.30	97.78*	47.14	41.05	43.77	42.64
DiffAttack		50.54	**46.62**	42.88	43.78	96.15*	69.45	75.01	67.66	56.56
AdvFM (ours)		**51.88**	39.69	**43.75**	**49.06**	**99.84***	**75.94**	**78.44**	**80.31**	**59.87**
AdvGAN	Swin-S	26.44	26.36	22.35	32.35	40.74	95.23*	43.02	49.66	34.42
AdvDiffuser		27.61	33.12	37.20	35.43	40.42	86.78*	44.54	42.42	37.25
DiffPGD		30.81	31.22	30.70	40.66	35.31	85.64*	44.88	35.99	35.65
AdvAD		40.50	46.38	32.00	35.22	43.57	97.46*	52.22	42.07	41.71
DiffAttack		36.53	**51.92**	**53.84**	46.15	60.30	98.30*	56.92	63.07	52.68
AdvFM (ours)		**56.56**	37.50	46.25	**49.69**	**80.00**	**98.91***	**75.63**	**75.63**	**60.18**

CNN-to-CNN attacks.

Overall, these results on CIFAR-10 mirror our ImageNet findings: injecting adversarial signals into the flow-matching velocity field and transporting them along the ODE produces perturbations that generalize well across architectures and inductive biases. The improvements are most pronounced for cross-family transfer (CNN→ViT and ViT→CNN), which is consistent with our theoretical analysis that AdvFM reduces gradient variance and aligns updates with a shared robust subspace rather than overfitting to a particular model.

9.1.3. Perceptual Quality Experiments

From Table 4, AdvFM achieves the best or second-best scores across all four perceptual metrics on ResNet34. In particular, AdvFM attains the highest PSNR and near-best SSIM, indicating that its adversarial examples stay very close to the clean images in pixel space while preserving local structures. At the same time, AdvFM achieves the lowest FID and a LPIPS score comparable to the strongest baseline (AdvAD), suggesting that the generated adversarial images remain visually natural in the feature space of deep perceptual networks. Qualitative results in Fig. 3 further confirm that AdvFM introduces minimal visible artifacts while still inducing misclassification, consistent with the goal of producing imperceptible yet highly transferable UAEs.

9.1.4. Attacking adversarially trained models

Table 5 reports ASR against four adversarially trained ResNet18 variants in the white-box setting. Classical generative and diffusion-based UAEs (AdvGAN, AdvDiffuser, AdvAD, DiffPGD, DiffAttack) achieve only moderate success: even the strongest baseline, DiffAttack, reaches at most 61.72% (Madry) and 60.19% (DeepMind), and often drops well below 50% on the more robust OA-AT model. In sharp contrast, AdvFM consistently attains very high ASR across all defenses, pushing success rates to 98.88%, 98.65%, 90.23%, and 85.45% on Madry, IAT, OA-AT, and DeepMind, respectively. These gains corroborate our theoretical analysis: injecting perturbations through the flow-matching velocity field yields directions that better align with robust gradients and concentrate in the robust–tangent subspace, enabling AdvFM to largely overcome the gra-

dient masking and off-manifold suppression induced by strong adversarial training.

9.2. Experiments on CelebA

9.2.1. Experimental setup

We additionally evaluate AdvFM on the CelebA face-attribute dataset to test whether the observed gains extend to another domain with different data statistics. We follow the standard CelebA split and train eight attribute classifiers: four CNNs (VGG16, ResNet34, DenseNet121, EfficientNet-B5) and four vision Transformers (ViT-B16, Swin-S, DeiT-S, VisFormer-S), implemented using `torchvision` and `timm`. Each model is used both as a surrogate (source) and as a black-box target. For a given source model, we generate unrestricted adversarial examples on a held-out test subset using AdvFM and all baselines under the same attack hyper-parameters as in the CIFAR-10 experiments, and report the attack success rate (ASR) on all remaining target models.

9.2.2. CelebA black-box transfer experiments

Tab. 6 shows that AdvFM achieves the highest average ASR for every source model on CelebA. When attacking CNN surrogates (VGG16, ResNet34), AdvFM substantially improves cross-architecture transfer. With VGG16 as source, the average ASR increases from 52.29% (DiffAttack) to 60.89%, with especially large gains on transformer targets such as Swin-S (66.88% vs. 54.40%), DeiT-S (67.81% vs. 42.40%), and VisFormer-S (68.75% vs. 48.91%). For ResNet34, AdvFM achieves an average ASR of 73.57%, far above the best baseline (49.08% for DiffAttack), and delivers strong transfer to all targets, including ViT-B16, Swin-S, DeiT-S, and VisFormer-S.

Using transformer surrogates (ViT-B16, Swin-S), AdvFM remains competitive on CNN targets and yields the strongest transfer to other transformers. With ViT-B16 as source, AdvFM obtains an average ASR of 59.87%, improving upon DiffAttack (56.56%) and reaching very high success rates on Swin-S (75.94%), DeiT-S (78.44%), and VisFormer-S (80.31%). Similarly, with Swin-S as source, AdvFM attains the best average ASR (60.18%) and clearly outperforms baselines on most targets, while maintaining solid transfer to CNNs.

Overall, the CelebA results mirror our findings on ImageNet and CIFAR-10: injecting perturbations into the flow-matching velocity field produces adversarial examples that transfer well across both convolutional and transformer-based face attribute classifiers. The gains are most pronounced for cross-family transfer (CNN$\rightarrow$ViT and ViT$\rightarrow$CNN), consistent with our theoretical analysis that AdvFM reduces gradient variance and aligns updates with a shared robust subspace rather than overfitting to a single architecture.