Generative AI Can Save Wildlife!

Saving Wildlife with Generative AI: Latent Composite Flow Matching for Poaching Prediction

Lingkai Kong*[1], Haichuan Wang*[1], Charles A. Emogor[1,2], Börsch-Supan[1], Lily Xu[3], Milind Tambe[1]
[1] Harvard University, [2] University of Cambridge, [3] Columbia University

Motivation: Wildlife poaching poses a major global threat, and predicting poaching is key to effective patrol planning.

Our Domain: Wildlife Conservation

Figure 1: Well-hidden snares and rangers conducting a patrol to locate them. Photos: Uganda Wildlife Authority

Problem Definition

We aim to predict poaching risk in protected areas using sparse ranger patrol data. Each park is divided into 1×1 km grid cells and monitored monthly. For each cell–month (i, t):

- $\mathbf{x}_{i,t}$: static and dynamic features (e.g., elevation, rainfall, vegetation)
- $a_{i,t,j}$: patrol effort on visit j (e.g., distance)
- $y_{i,t,j}$: binary detection (poaching observed or not)

Poaching is a latent binary variable $z_{i,t} \in \{0, 1\}$, unobserved unless detected. Detection is imperfect and depends on patrol effort.

Our goal is to estimate the poaching risk $p(z_{i,t} = 1)$ based on historical observations and patrol behavior.

Background on Flow Matching (FM)

Flow Matching learns a time-dependent velocity field $v_\theta(\psi^{(s)}, s)$ that transports samples from a source distribution p_0 at $s = 0$ to a target distribution p_1 at $s = 1$. Inference is performed by sampling from p_0 and integrating the learned ODE. **Conditional Flow Matching** extends FM by conditioning the velocity field v_θ on context, enabling flows to adapt to varying environmental conditions.

Overview of WildFlow

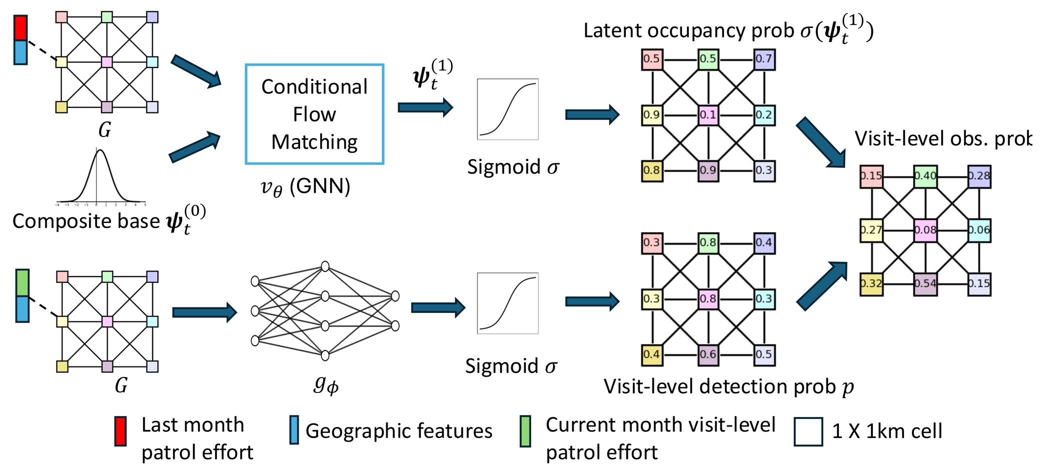


Figure 2: Overview of **WildFlow**. **Upper branch** A composite base initializes latent logits $\boldsymbol{\psi}_t^{(0)}$ from a pretrained linear occupancy model with Gaussian noise. A graph conditional velocity field v_θ transports $\boldsymbol{\psi}_t^{(0)}$ to $\boldsymbol{\psi}_t^{(1)}$ via FM. **Lower branch** the visit level detector uses geospatial features and current month visit effort to predict detection probabilities. Results from both branches are combined to compute the occupancy-detection likelihood.

Model Components

- **Occupancy Model → Imperfect Detection**
  We model the true presence of snares as a latent occupancy probability, and define the visit-level detection probability as conditional on this latent state.
- **Latent Composite Flow → Data Scarcity**
  Train a composite flow framework warm-starting from a base distribution constructed from the prediction of a linear occupancy model.

Training

- Stage 1: Encoder-detector training
  - an encoder that predicts the surrogate latent occupancy logits
  - a detection head that predicts visit-level detection probabilities

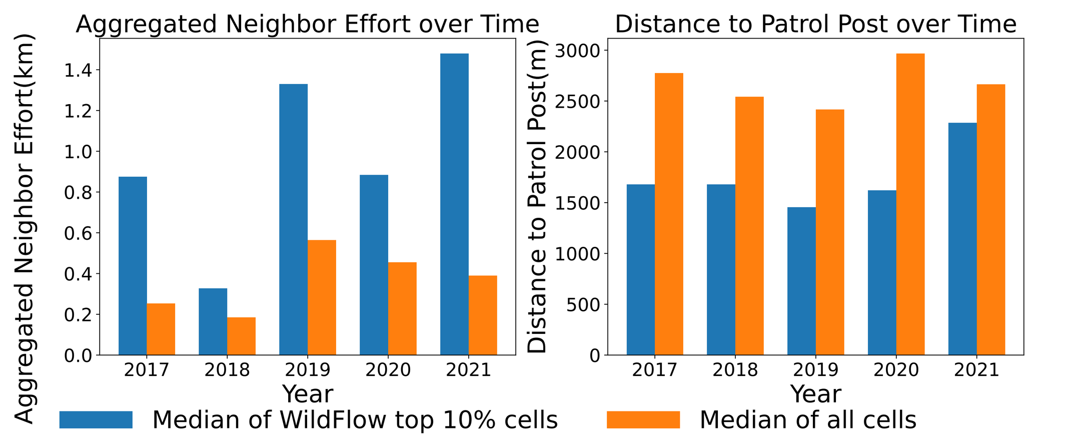
  We jointly train two components which maximize the occupancy-detection likelihood.
- Stage 2: Latent flow training
  We freeze the encoder-detector pair and train a conditional flow model v_θ to transport samples from base to the surrogate latent logits.

Experiment Results

Park	Year	LogReg	GP	MLP	GNN	Transformer	Diffusion	Ours
MFNP	2017	0.305	0.278 ± 0.001	0.288 ± 0.017	0.336 ± 0.004	0.314 ± 0.016	0.313 ± 0.021	**0.374 ± 0.032**
	2018	0.344	0.308 ± 0.002	0.325 ± 0.004	0.260 ± 0.000	0.302 ± 0.064	0.279 ± 0.001	**0.377 ± 0.011**
	2019	0.406	0.359 ± 0.002	0.388 ± 0.007	**0.526 ± 0.015**	0.475 ± 0.141	0.362 ± 0.003	0.409 ± 0.009
	2020	0.421	0.401 ± 0.003	0.408 ± 0.004	0.366 ± 0.014	0.327 ± 0.002	0.439 ± 0.023	**0.473 ± 0.006**
	2021	0.360	0.364 ± 0.011	0.335 ± 0.010	**0.430 ± 0.023**	0.295 ± 0.085	0.342 ± 0.032	**0.423 ± 0.004**
	Avg	0.367	0.342	0.349	0.384	0.343	0.347	**0.411**
QENP	2014	0.119	0.107 ± 0.003	0.103 ± 0.003	**0.136 ± 0.013**	0.100 ± 0.002	0.102 ± 0.003	0.116 ± 0.009
	2015	0.180	0.156 ± 0.002	0.172 ± 0.006	0.111 ± 0.000	0.173 ± 0.016	0.162 ± 0.028	**0.201 ± 0.017**
	2016	0.258	0.220 ± 0.005	0.227 ± 0.004	0.238 ± 0.009	0.220 ± 0.019	0.201 ± 0.034	**0.299 ± 0.027**
	Avg	0.186	0.161	0.167	0.162	0.164	0.155	**0.205**

- WildFlow outperforms the strongest baselines in AUPR by an average of 7% and 10% in the two parks, respectively.
- Linear model remains competitive on small or noisy poaching data due to less overfitting.
- Diffusion model underperforms, showing the importance of an informative initial distribution.

Case Study



- Largest gains occur near areas with high patrol effort, highlighting WildFlow's strength in modeling displacement-driven dynamics.
- WildFlow better exploits proximity to patrol posts, likely benefiting from reliable detections.

Takeaway

- The first generative AI framework for poaching prediction.
- Deployable for real-world conservation efforts.

Acknowledgements

We thank the Uganda Wildlife Authority for data access from Murchison Falls and Queen Elizabeth National Parks. This work was supported by ONR MURI N00014-24-1-2742.

Scan to read the full paper