(Machine) Learning to create artwork and quantum fields

Pavel Buividovich (University of Liverpool)

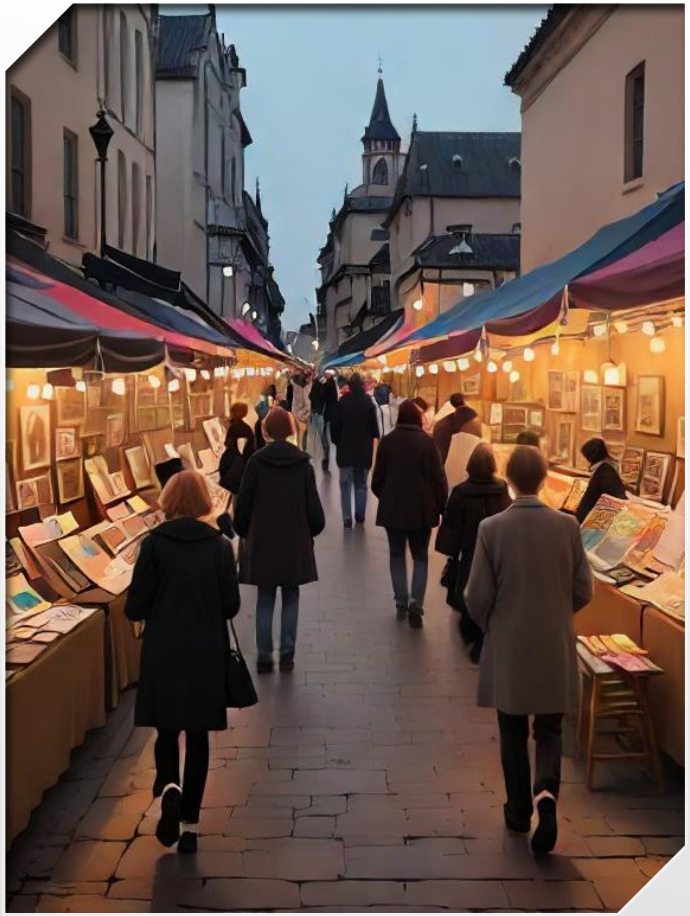
Work in collaboration with: Art Recognition AG (Carina Popovici, Ludovica Schaerf), Eric Postma (Tilburg University), Johann Ostmeyer (Bonn University), Joseph Hadley (UoL)

Based on Ostmeyer, Schaerf, Buividovich, Charles, Postma, Popovici, **Synthetic images aid the recognition of human-made art forgeries**. PLOS ONE 19(2): e0295967. https://doi.org/10.1371/journal.pone.0295967

Talk outline

- **Setting the stage: Artefact/forgery detection using ML**
- **Generative neural networks for image generation**
- **Adversarial learning**
- **Quantum field configurations vs artwork**
- **Monte-Carlo algorithms and their challenges**
- **Using neural networks to accelerate Monte-Carlo (normalizing flow)**

- Global art market worth ~ $60 billion
- Authenticity/value of assets established by expert opinions
- Selling artwork forgeries is a lucrative criminal activity



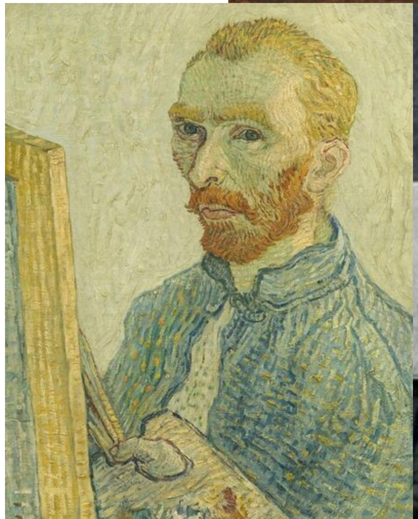
Forgeries and art market

Amede-No! Modigliani Show Shut Down After 21 Works Deemed Likely Fakes

The exhibition, at the Doge's Palace in Genoa, included pieces on loan from private collections and major institutions like the Musée de l'Orangerie and the Fitzwilliam Museum.

Forgeries and art market

- Wolfgang Beltracchi forged many artworks by famous authors
- In 2006, Beltracchi's fake "La Horde" (assumed author Max Ernst) sold at Christie's for £3,000,000
- In 1920s, Otto Wacker sold >30 fake Van Gogh paintings, of which many were included in catalogues
- John Myatt, British author of "genuine forgeries"
- Many forgeries are not discovered yet …



Art classification and machine learning

- Human expert opinions often contradict each other
- Machine learning methods: more objective decisions?
- Most studies to date concentrate on artwork attribution
- Style extraction and attribution algorithms:
  - fractal analysis
  - wavelets
  - sparse coding
  - clustering-based segmentation
  - tight frame method
  - Convolutional neural networks (CNNs)
  - Visual transformer NNs

Challenges of art authentication

- CNNs: supervised learning, trained on labelled datasets
- Default option: Van Gogh vs everything not Van Gogh – not very useful (all forgeries are still Van Gogh)
- Better: Binary classification, Van Gogh vs all known forgeries of Van Gogh
- Challenge: 900 paintings + >1000 sketches/drawings by real Van Gogh, < 50 known forgeries (30 by Wacker)
- Huge imbalance in True/False datasets, typical for most well-known artists

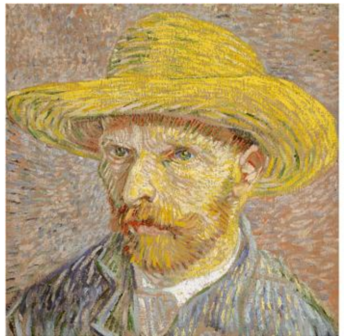



Image by wirestock on Freepik/Image by master1305 on Freepik

Art forgeries and Generative AI

- Modern GenAI able to learn any artist's style nowadays
- GenAI can create advanced art forgeries
- Artefacts quite different from human ones (evidenced by Fourier analysis etc), can be removed once known
- Considered a threat to art/creativity market and artists' jobs
- Let's turn things around and use GenAI to protect art market!

MIT Technology Review

Featured Topics Newsletters Events Podcasts SIGN IN SUBSCRIBE

ARTIFICIAL INTELLIGENCE

Artists can now opt out of the next version of Stable Diffusion

The move follows a heated public debate between artists and tech companies over how text-to-image AI models are trained.

Billie Eilish, Nicki Minaj, Stevie Wonder and more musicians demand protection against AI

Letter signed by more than 200 artists makes broad ask that tech firms pledge to not develop AI tools to replace human creatives

Generative AI and art authentication

Work with **Art Recognition AG** (Carina Popovici, Eric Postma, Ludovica Schaerf) and Johann Ostmeyer (Bonn):

- use GenAI to create more balanced datasets for art authentication

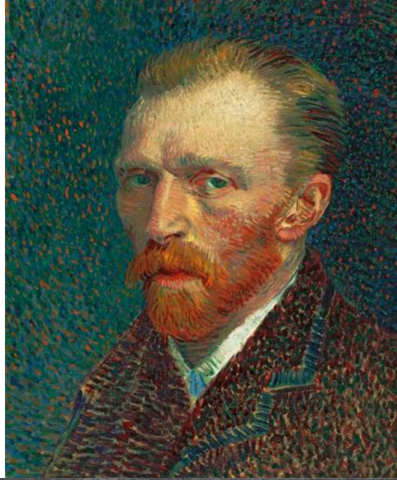
- I'll cover technical solutions behind this work before moving to quantum physics

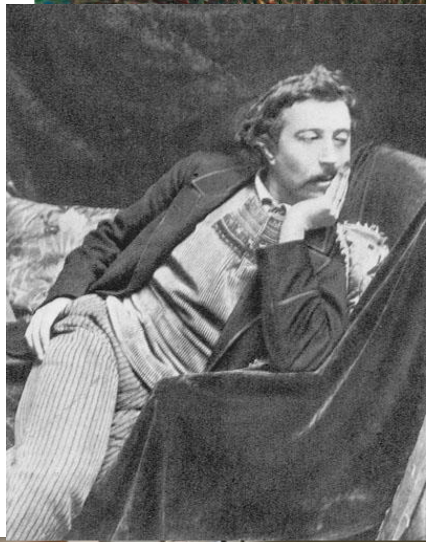

ART RECOGNITION

Dataset composition

- Labelled data/supervised learning:
  - 126 original Van Gogh paintings (RGB) => Ground truth
  - 212 stylistically similar images (other impressionist/expressionist artists, van Gogh followers) => Contrast set (mostly for pre-training)
  - 11 Wacker forgeries => Contrast set
  - 8 “genuine forgeries” by John Myatt => Contrast set

- Authenticity analysis mainly based on small-scale details
- Use patches of original images
- 21, 5, or 1 adjacent non-overlapping patches.
- Bi-cubic resampling to 224 × 224 or 256 × 256 (classifier inpu
- Split patches into training (72%), validation (11%), and test (17%) sets
- 10 random splits, bootstrapped cross-validation



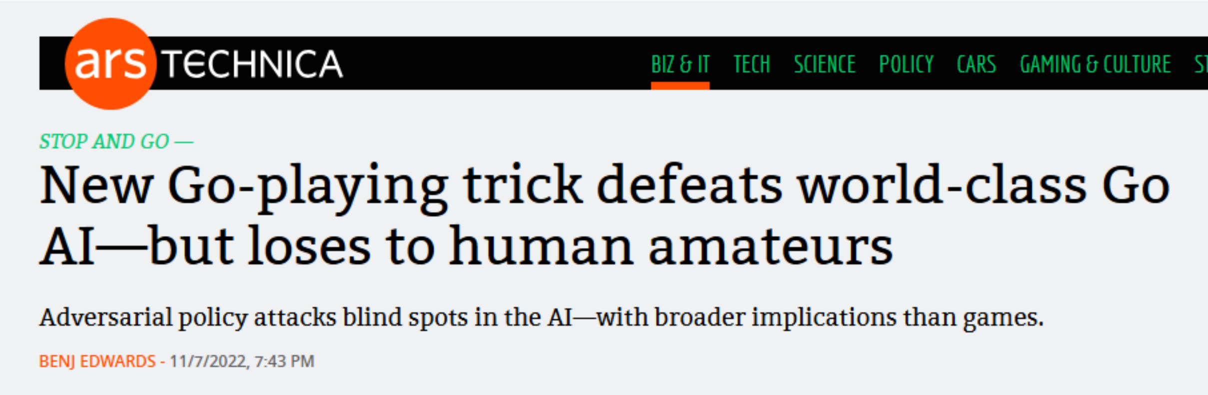


Generative adversarial networks (GANs)

[Goodfellow and collaborators’2014]

- Zero-sum game: generator vs. discriminator (win of one is the loss of another)
- Discriminator *D(x)*: image *(x)* -> *[0 … 1]* (authentic/not authentic)
- Generator *G(z)*: latent space *(z)* -> image *G(z)*
- Cost function:

$$\mathrm{Cost}(D, G) = \langle \log\left(D(x)\right)\rangle_{\mathrm{data}} + \langle \log\left(1 - D\left(G(z)\right)\right)\rangle_{\mathrm{generator}}$$


ars TECHNICA

BIZ & IT TECH SCIENCE POLICY CARS GAMING & CULTURE

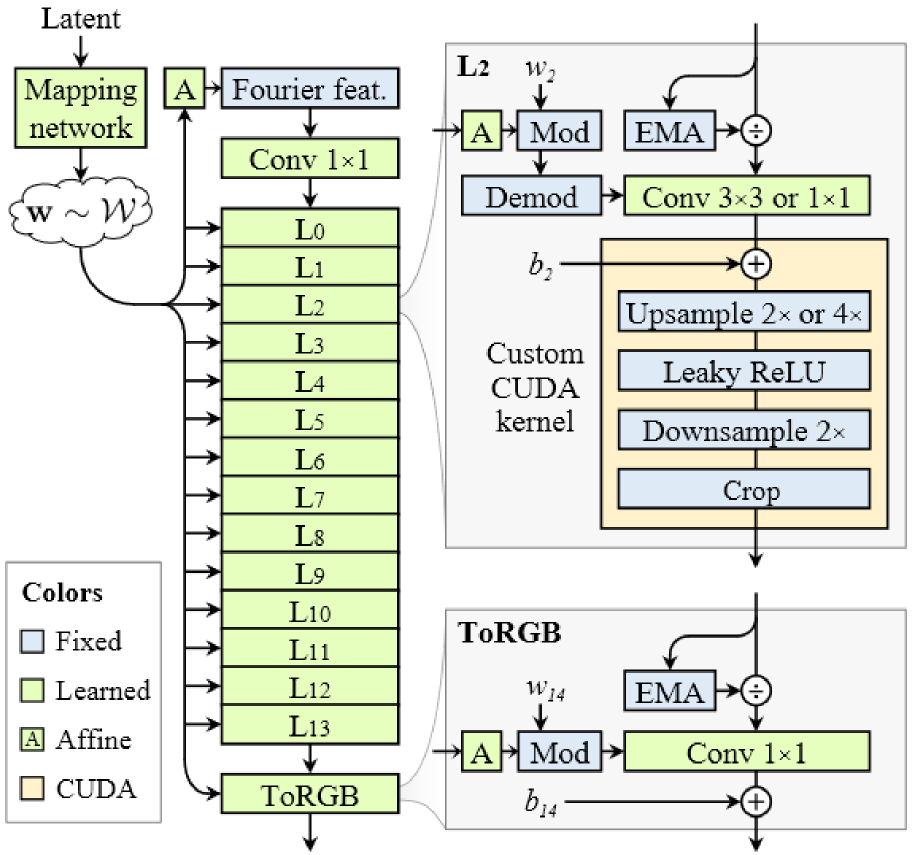
STOP AND GO —

New Go-playing trick defeats world-class Go AI—but loses to human amateurs

Adversarial policy attacks blind spots in the AI—with broader implications than games.

BENJ EDWARDS - 11/7/2022, 7:43 PM

StyleGAN



From [**Alias-Free Generative Adversarial Networks,**
Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, Timo Aila, ArXiv:2106.12423]

- Mapping network: Latent space -> latent code
- L0 … L13 flexible layers operate in Fourier space
- Increasing frequency cutoff to allow for finer and finer details
- Each layer receives random input (bias b_2, linear transform w_2)

StyleGAN training and tuning

Image by tohamina on Freepik

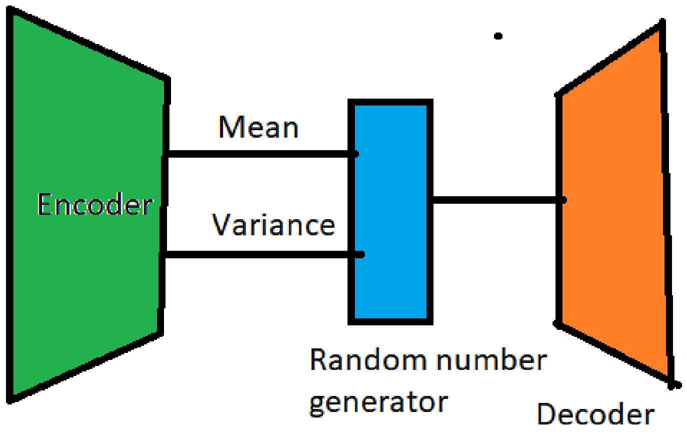
- **Pre-training**: 10380 portraits in all genres
- Many different authors (including Van Gogh)
- 5M epochs on 4 GPUs
- **"Raw GANs" dataset**: random synthetic portraits in a variety of styles
- **Tuning:** 50k epochs training on Van Gogh originals only
- **"Tuned GANs" dataset:** synthetic Van Gogh forgeries
- Longer tuning results in overfitting, StyleGAN mainly reproduces training data
- (20k – 100k epochs enough to learn the author style and avoid overfitting)

StyleGAN training and tuning

Image by tohamina on Freepik

- We use default settings for StyleGAN2
- Works better for artwork than the more advanced StyleGAN3 (optimized for photorealistic images/video)
- StyleGAN3 improves translational invariance
- Tends to smear local hard transitions, often featured by brush strokes

Variational Autoencoders (VAEs) (prelude to stable diffusion)



- $q_\phi(z|x)$ – Encoder, data (x) → latent space (z)
- Gaussian models (most often)

$$q_\phi(z|x) = \mathcal{N}(\mu(x), \sigma^2(x)))$$

- $p_\theta(x|z)$ – Decoder, latent space (z) → data (x)
- z → x, x → z : approximated in terms of the deep neural network with parameters θ and ϕ
- Cost function: Evidence Lower Bound (ELBO)

$$L_{\theta,\phi} = \langle \log(p_\theta(x|z)) \rangle_{z\sim q_\phi(z|x))} - D_{KL}\left(q_\phi(z|x)|p(z)\right)$$

Likelihood of reconstructed data

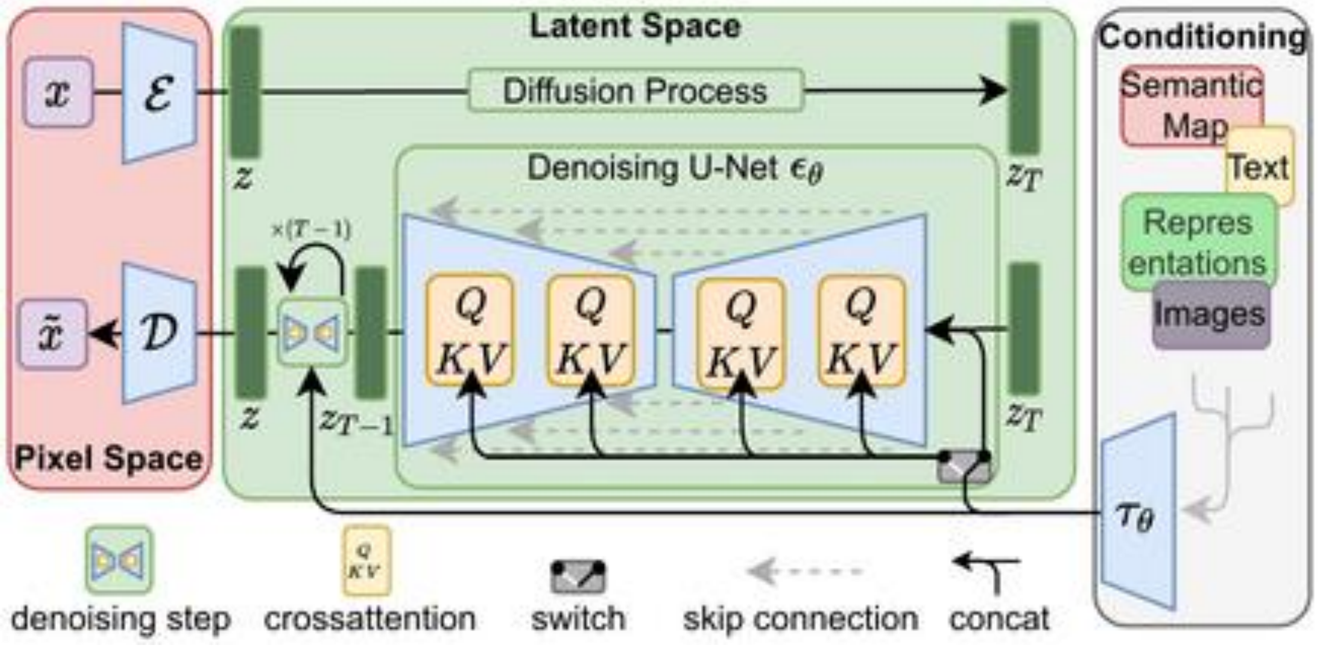
Deviation of $q_\phi(z|x)$ from unit Gaussian $p(z)$

Kullback–Leibler divergence

$$D_{KL}(p_1(y)|p_2(y)) = \left\langle \log\left(\frac{p_1(y)}{p_2(y)}\right) \right\rangle_{p_1}$$

- D_{KL} prevents $q_\phi(z|x)$ from learning the data exactly

Stable diffusion



- We use Stable Diffusion 2.0 as is
- No post-training or fine-tuning
- The model is already trained on a huge amount of data

- Variational Autoencoder: learns the latent space representation of images and generates output images
- Denoising: transform/de-noise latent space conditioned on text prompt/other image/etc.

Output data

Original

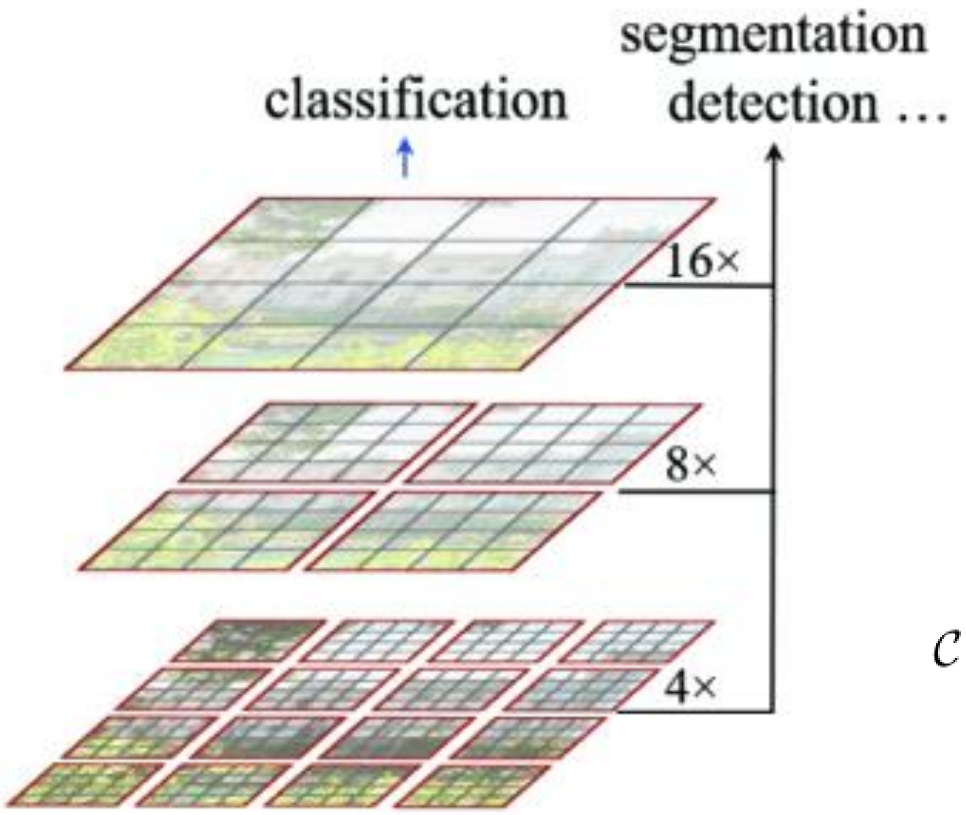
Wacker forgery

StyleGAN

Stable Diffusion

- Two different classifiers to recognize forgeries
- Not a competition, the goal is to demonstrate universality

Forgery detection: transformer-based classification (SwinBase)



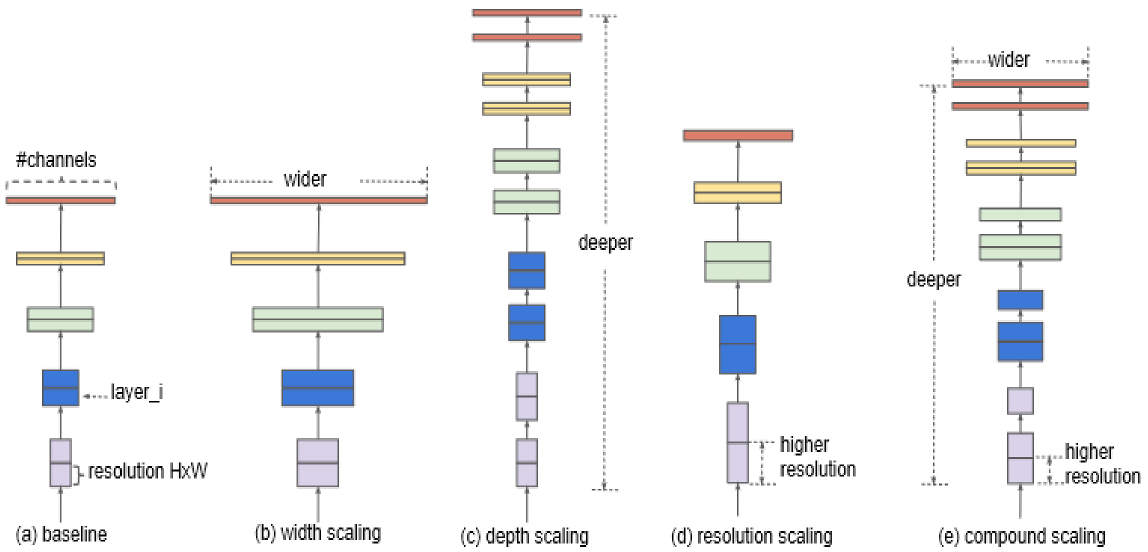
From Liu Z, Lin Y, Cao Y, Hu H, Wei Y, Zhang Z, et al. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), https://dx.doi.org/10.1109/ICCV48922.2021.00986.

- Analysis of hierarchical feature maps
- 224 x 224 x RGB input, 88M parameters
- Final activation layer → dense layer converging in a sigmoid
- Binary classification
- Cost function: binary cross-entropy

$$\mathcal{C} = -\sum_i y_i \log\left(P\left(y_i = 1 | x_i\right)\right) - \sum_i \left(1 - y_i\right) \log\left(P\left(y_i = 0 | x_i\right)\right)$$

- Learning rate 10^{-5}, batch size 32

Forgery detection: CNN classification (EfficientNet B0)



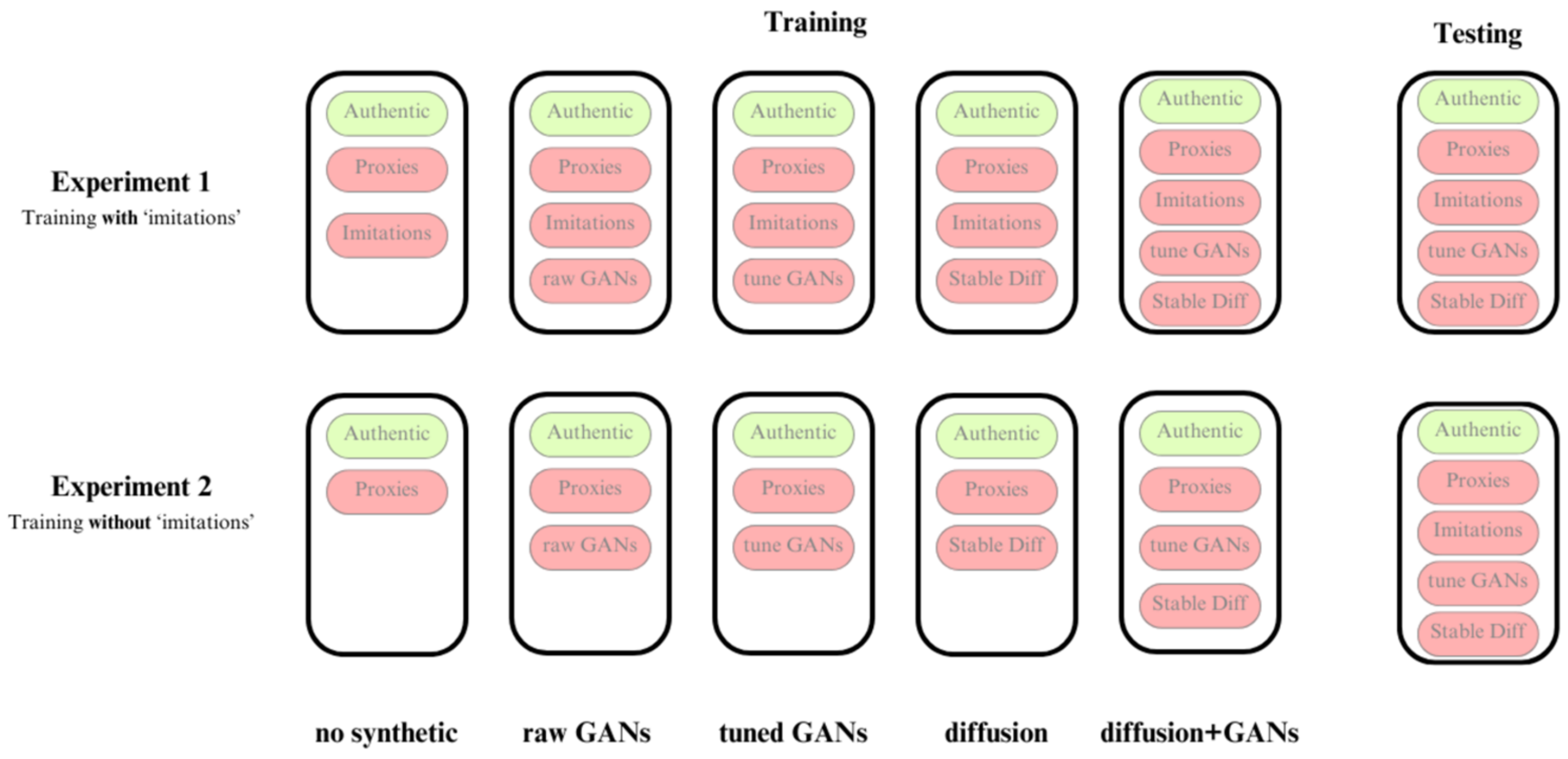
- 256 x 254 x RGB input, 5.3M parameters

- Binary classification
- Cost function: binary cross-entropy

- Learning rate 10^{-5}, batch size 32

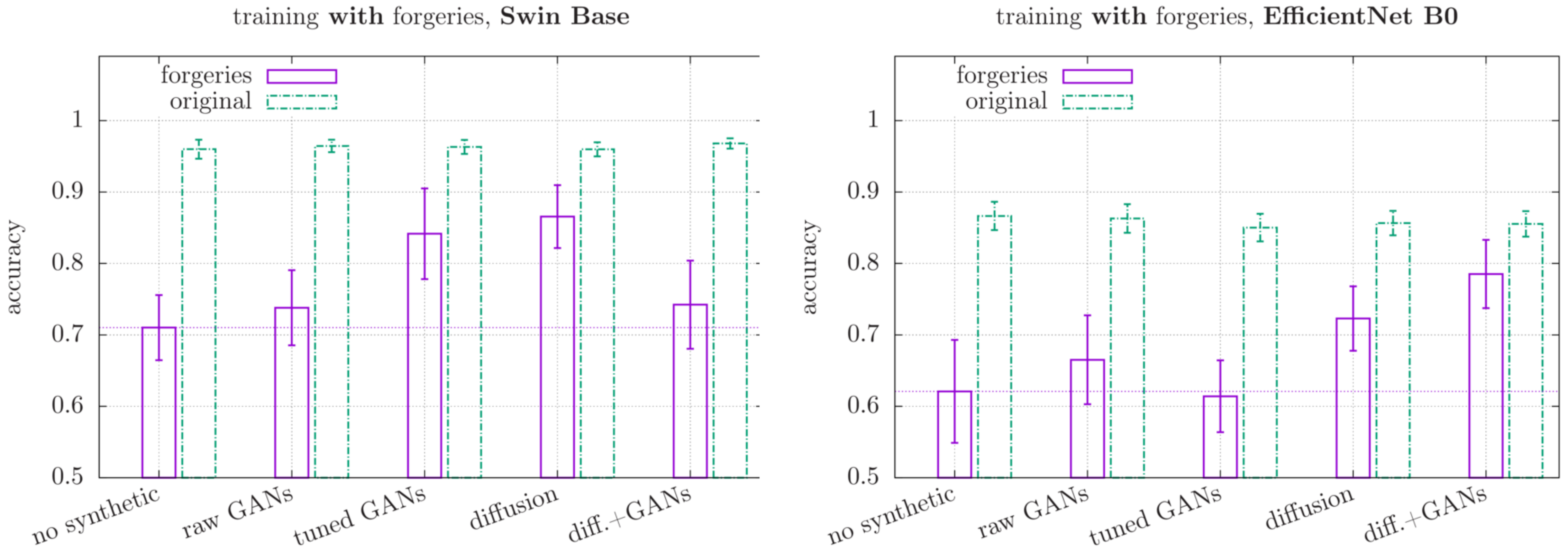
Tan M, Le Q. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. In: Chaudhuri K, Salakhutdinov R, editors. Proceedings of the 36th International Conference on Machine Learning. vol. 97 of Proceedings of Machine Learning Research. PMLR; 2019. p. 6105–6114. Available from: https://proceedings.mlr.press/v97/tan19a.html.

Experiment setup



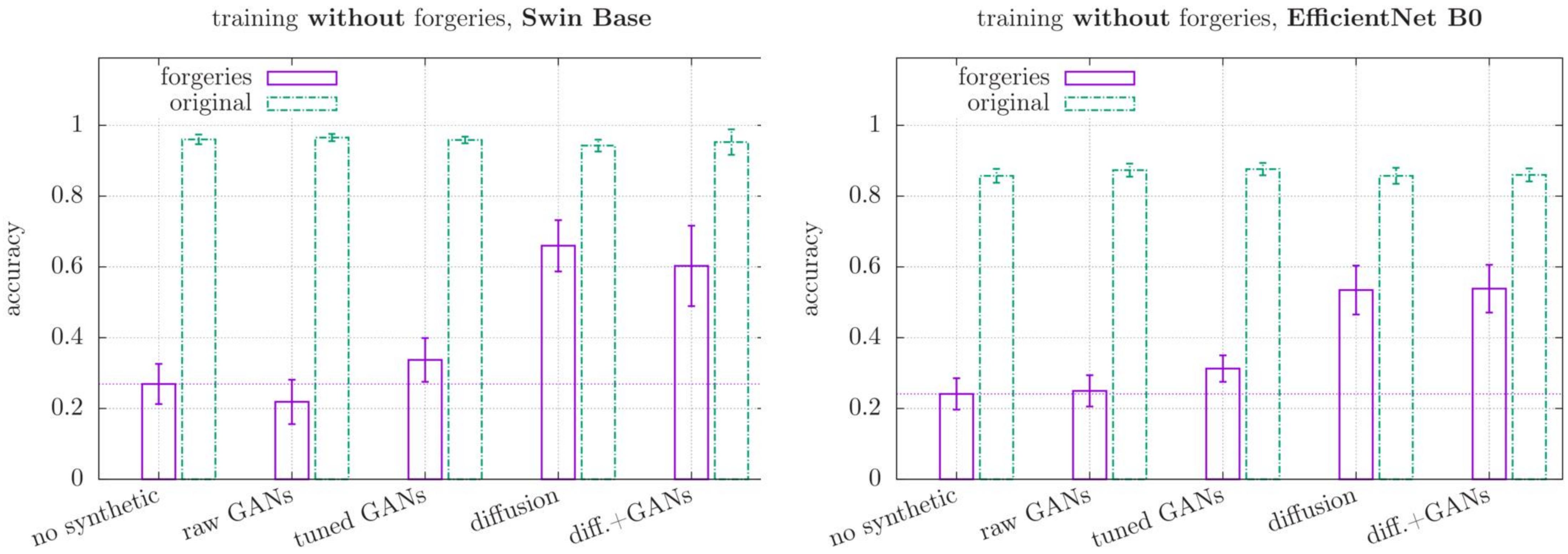
- How synthetic forgeries help to detect human-made ones?
- 30 synthetic images → 150 patches in each category
- Can GenAI replace human-made forgeries altogether? (If there are no known forgeries at all)

Results – human forgeries + synthetic



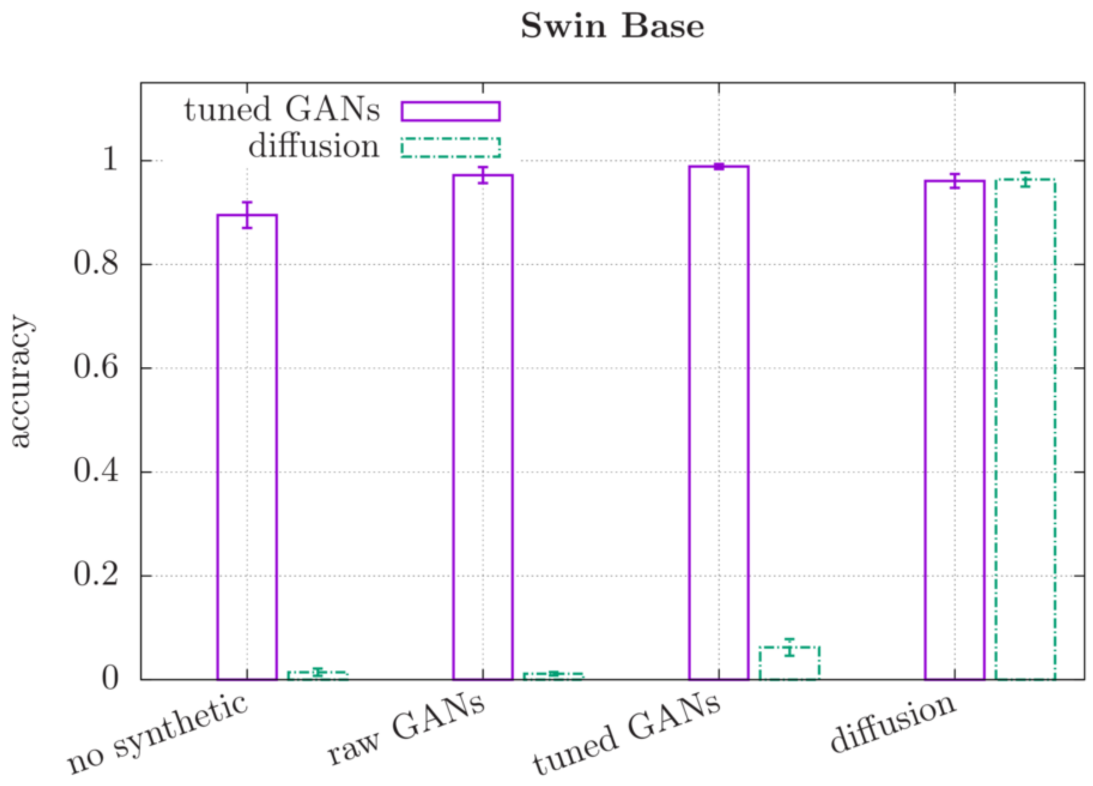
- GAN images alone may or may not help depending on classifier
- Stable diffusion always improves accuracy
- Too good data results in overfitting (diff.+GANs for Swin Base)

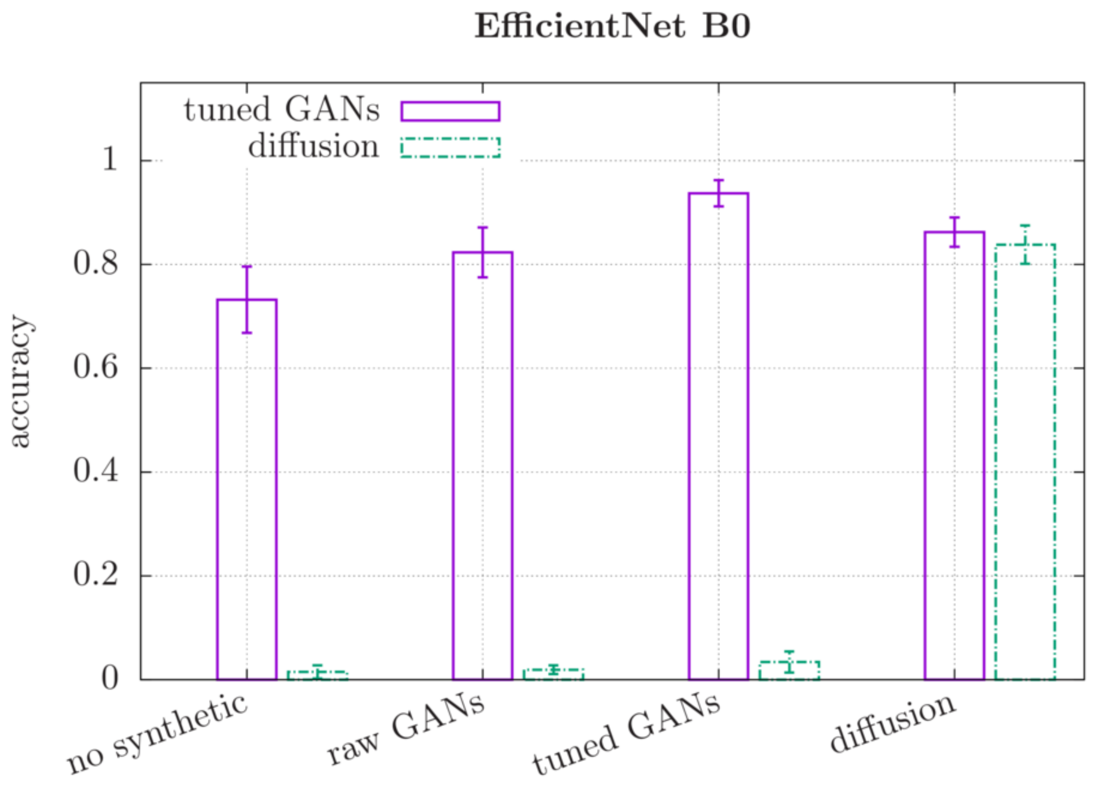
Results – synthetic only



- Here the model never sees human forgeries during training
- Again, Stable diffusion always improves accuracy
- Too much synthetic data makes classifier primarily detect GenAI results
- Success of Stable Diffusion? Sheer amount of training data?

Results – detecting synthetic data

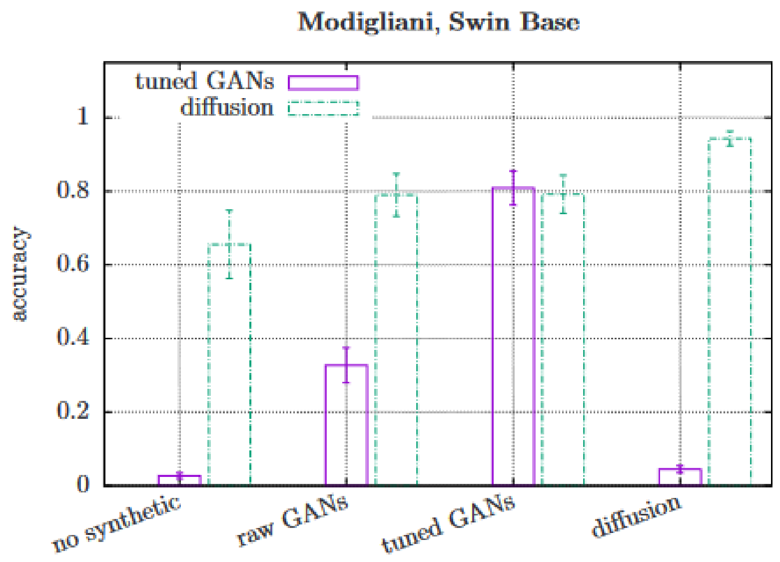


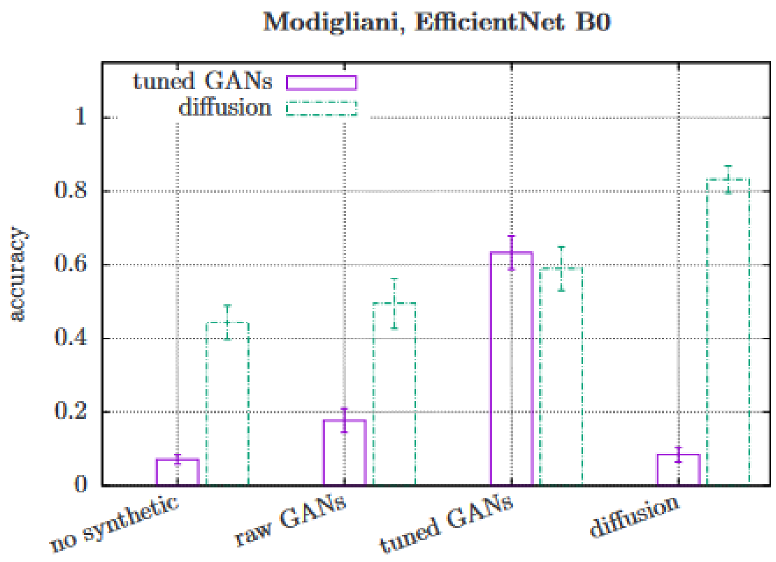


- Classifiers are trained on 4 different datasets (incl/excl. synthetic data)
- Tested on unseen synthetic data – either GANs or Stable Diffusions
- GANs appear much easier to detect even when previously unseen
- Partially explains the success of Stable Diffusion

Outlook

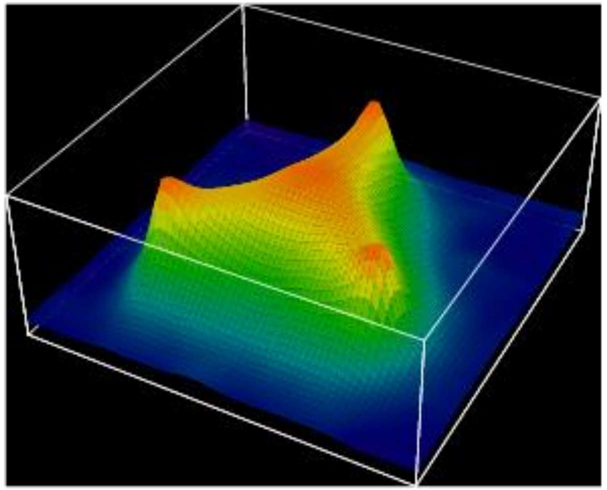
- It appears that Stable Diffusion is the best "AI forger"
- Results are author-dependent: e.g. no particular advantage of Stable Diffusion for Modigliani
- All tuned synthetic data improves accuracy of human forgery detection
- Unsupervised learning approaches?
- Inclusion of more data layers (e.g. chemical composition)?

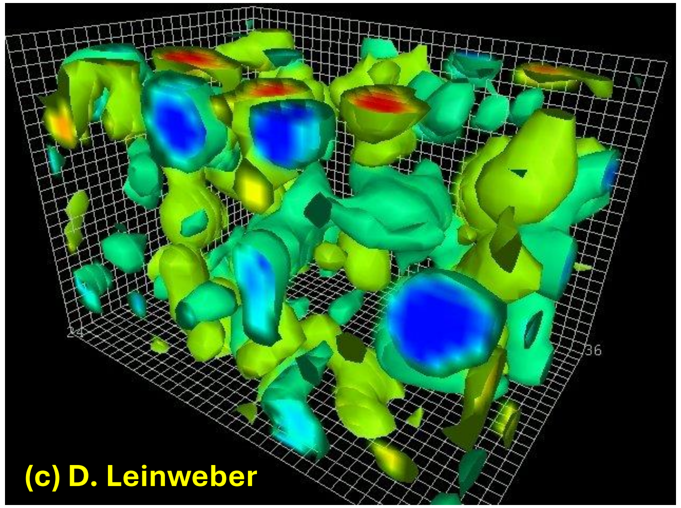




- **“Mother Nature is the greatest artist and water is one of her favorite brushes.”** — Rico Besserdich, underwater photographer

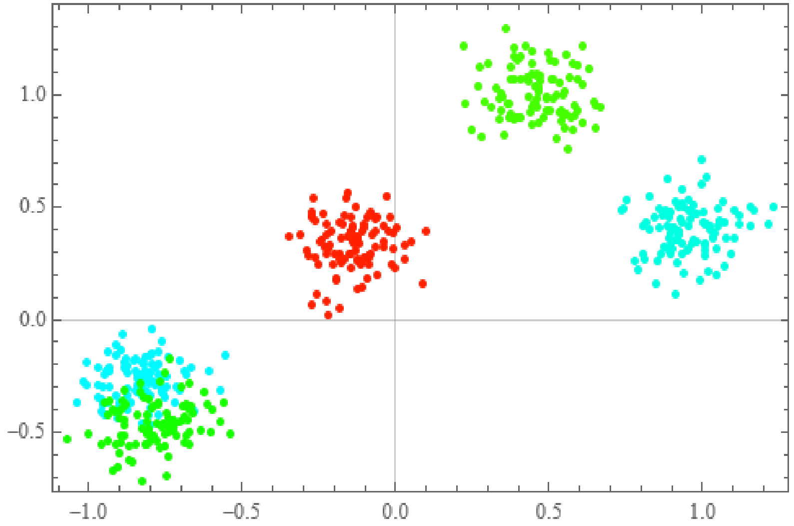
- As a theoretical physicist, I’d say **quantum fields** are Nature’s favorite brushes...

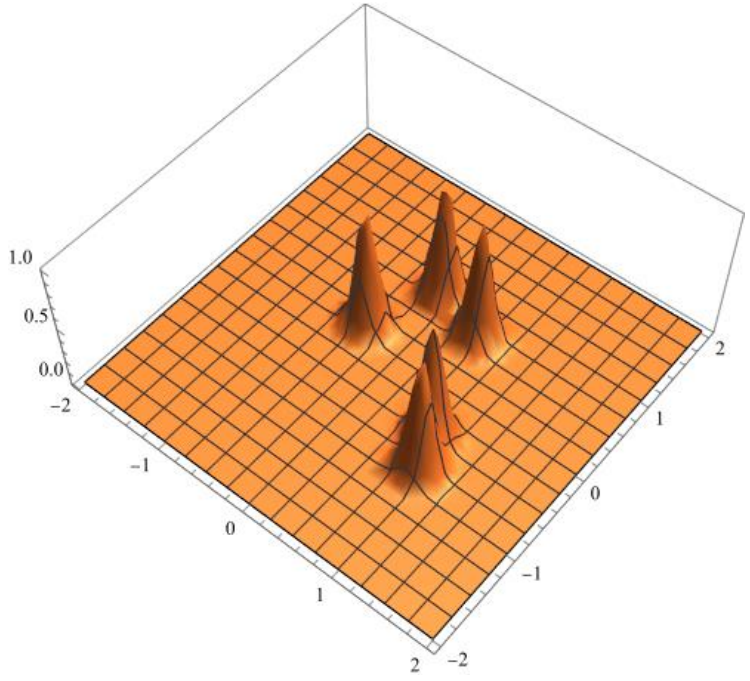




- What I discuss further applies equally to **statistical physics**

Probabilistic/Bayesian interpretation of GenAI





- We are trying to learn a probability distribution *p(x)* of data *x*
- Any artist only produces a finite amount of works
- Probability distribution is just a collection of Dirac delta-functions
- No infinite statistical ensemble exist
- Approximate with a smooth, continuous distribution
- This distribution is complex and multi-modal

Probabilities in statistical physics

- Probability distribution is usually known exactly as a simple* mathematical formula
- Ising model ($\sigma_x = \pm 1$):

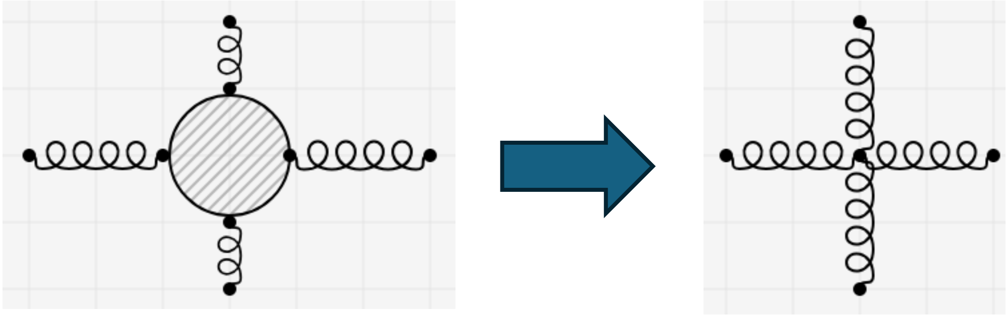
$$P\left[\sigma_x\right] = \mathcal{N} \exp\left(\beta \sum_{<x,y>} \sigma_x \, \sigma_y\right)$$

  - Large number of degrees of freedom
  - Emergent complex phenomena (percolation, fractality)
  - Despite mathematical simplicity:
  - Multi-modal probability distributions, regions of large weight interleaved with almost empty areas

Probabilities in quantum field theory

- Quantum amplitudes/partition functions written as path integrals
- QCD, the theory of strong nuclear interactions

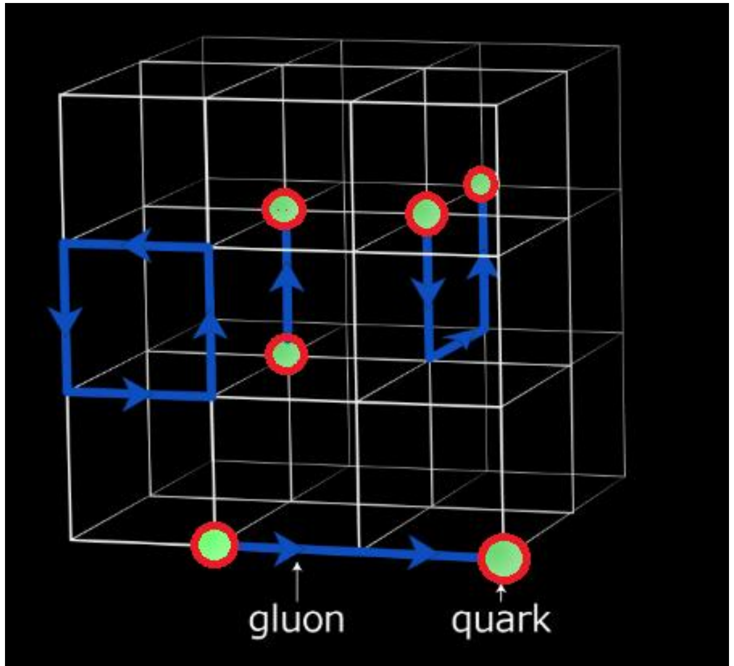
$$\mathcal{Z} = \int \mathcal{D}A_\mu(x)\mathcal{D}\bar{q}(x)\mathcal{D}q(x)$$
$$\exp\left(-\int d^4x\left(\bar{q}\gamma^\mu\left(\partial_\mu - igA_\mu\right)q + \mathrm{Tr}\,F_{\mu\nu}^2\right)\right)$$



- Probability only properly defined for bosonic fields (gluons $A_\mu(x)$)
- Fermionic fields: anticommuting, only make sense in integrals
- Complex, nonlocal weight for $A_\mu(x)$ after integrating out $q(x)$

$$\mathcal{Z} = \int \mathcal{D}A_\mu(x)\det\left[\gamma^\mu\left(\partial_\mu - igA_\mu\right)\right]^{N_f}\exp\left(-\int d^4x\,\mathrm{Tr}\,F_{\mu\nu}^2\right)$$

Lattice field theory

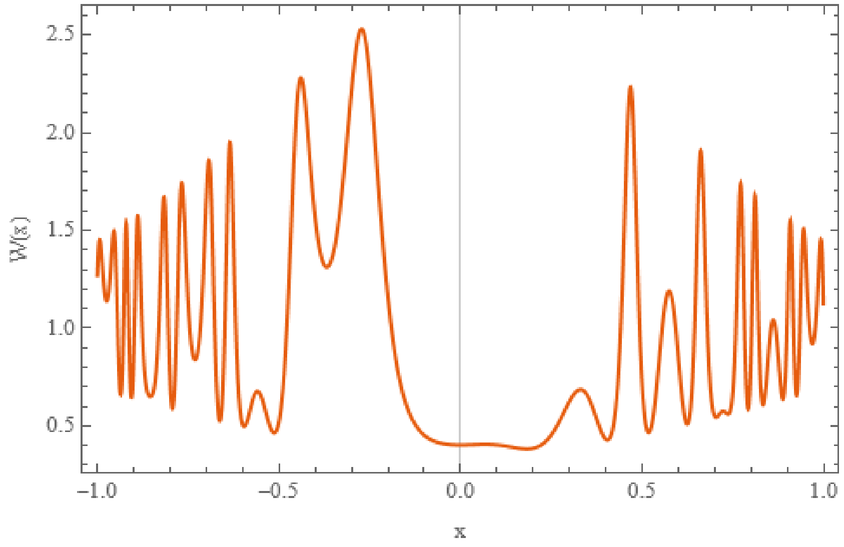


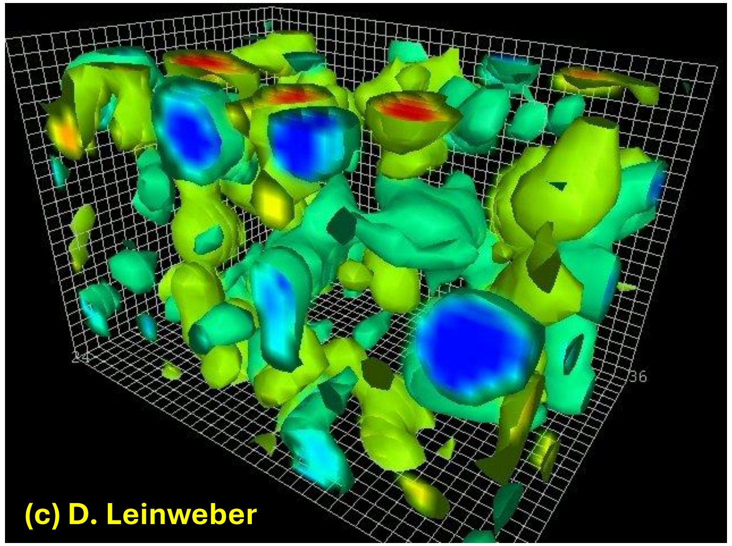
- Continuous coordinates → lattice (space + time or just space)
- Scalar fields → Lattice sites
- Vector fields → Lattice links
- Rank-2 tensors → Lattice plaquettes

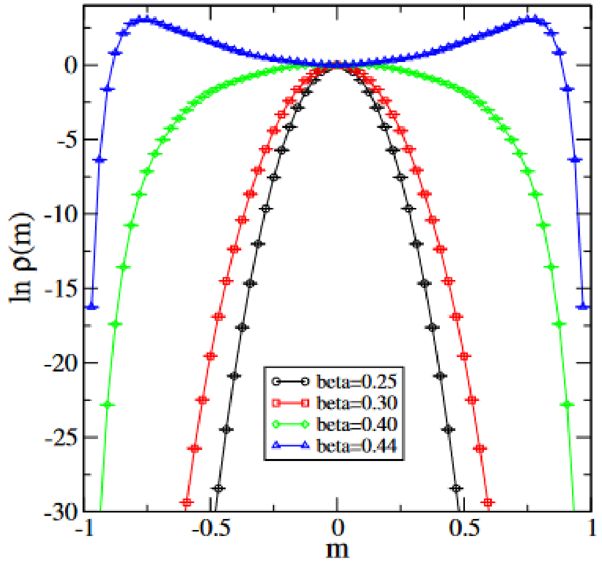
- Infinite-dimensional path integrals → high-dimensional ordinary integrals
- Integral weight ~ Probability
- Sampling via Monte-Carlo

Probabilities in quantum field theory

- QCD path integral weight features (almost) disjoint topological charge sectors







- Sectors of positive/negative magnetizations in the Ising model
- Problem: How to generate configurations according to very high-dimensional, unfactorizable probability?

Monte-Carlo sampling and Metropolis algorithm

- Set of stochastic updates $Y \rightarrow X$, probability $P(X|Y)$
- Target probability distribution $W(X)$
- Accept each update with probability

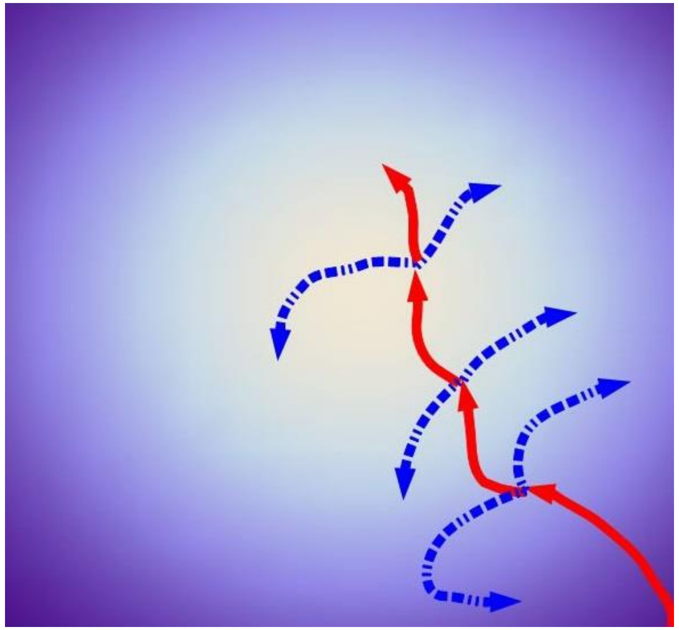
$$A\left(X|Y\right) = \min\left(1, \frac{W\left(X\right)}{W\left(Y\right)} \frac{P\left(Y|X\right)}{P\left(X|Y\right)}\right)$$

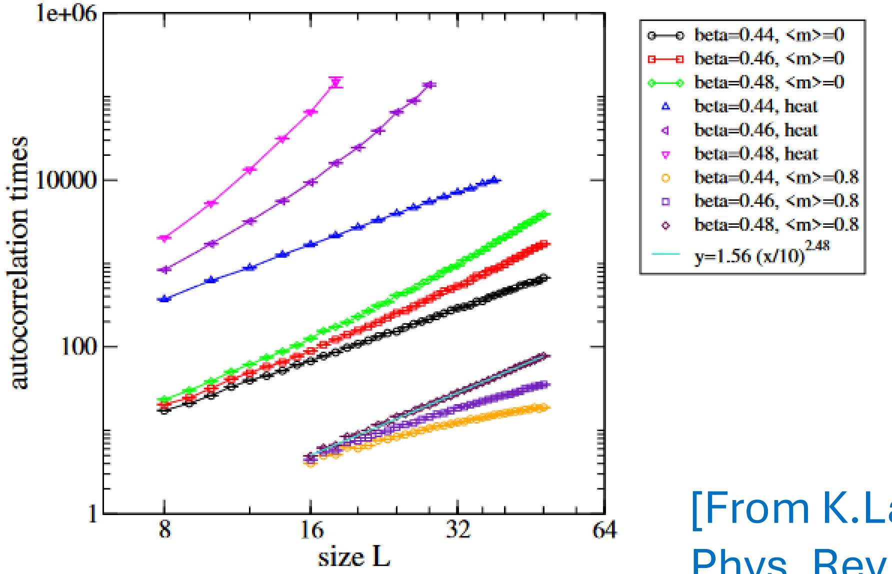
- Updates should be ergodic (any X reachable from any Y via a finite number of updates)
- Good updates have high acceptance probability
- They are notoriously difficult to design
- Example: cluster updates vs. local updates for the Ising model

Metropolis algorithm and autocorrelations

- With conditional updates $Y \rightarrow X$, X and Y are correlated
- Monte-Carlo averaging needs statistically independent samples
- Decorrelating samples may take many updates – measured in terms of autocorrelation times ...

$$\langle \mathcal{O}(X_i)\,\mathcal{O}(X_{i+k})\rangle - \langle \mathcal{O}(X_i)\rangle\langle \mathcal{O}(X_{i+k})\rangle \sim \exp\left(-k/\tau_a\right)$$





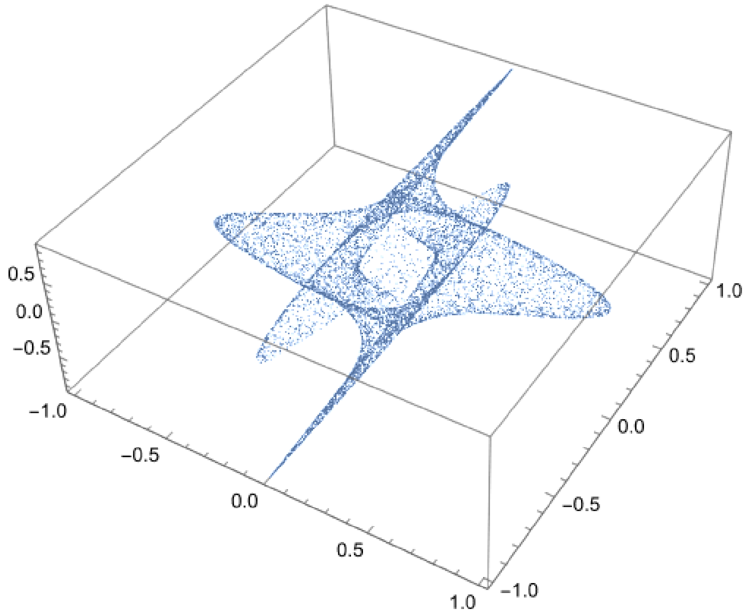
Autocorrelation times grow exponentially with volume

Mathematically, an NP-hard problem!

[From K.Langfeld, PB, P.Rakow, J.Roscoe Phys. Rev. E 106, 054139]

Machine learning approaches to Monte-Carlo

- Instead of devising update schemes ourselves, can we use ML to learn the required probability distribution?
- GenAI approaches that work well for images/text/music/etc. do not straightforwardly generalize to Monte-Carlo…
- Low-dimensions latent space is enough to capture the essential info in images etc…
- Always a complex hypersurface embedded in a high-dimensional configuration space
- Not ergodic if we want to sample the entire configuration space



Normalizing flow

- Let's remember how to sample from an arbitrary 1D probability distribution *p(x)*...

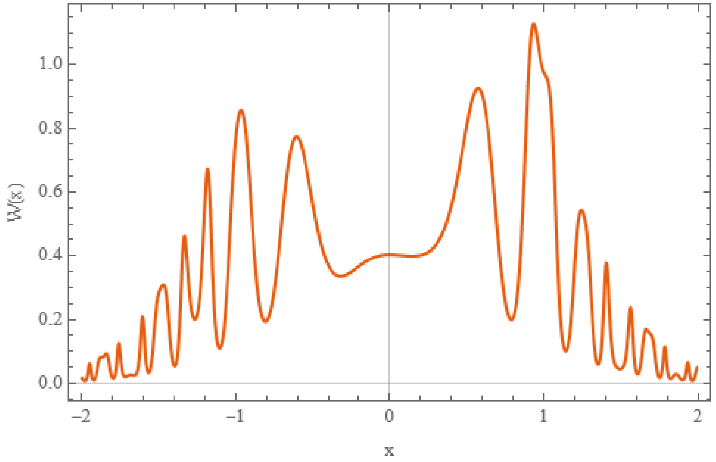
$$\Phi(x) = \int_{-\infty}^{x} dy P(y) \in [0,1]$$

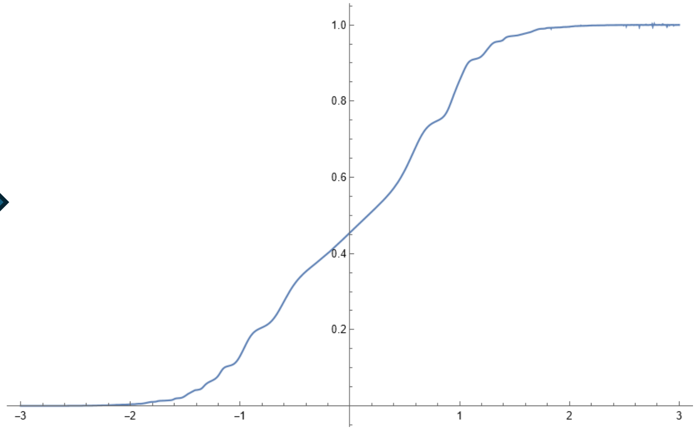
- Random $q \in [0, 1]$ $\quad x = \Phi^{-1}(q)$ - inverse function, normalizing flow

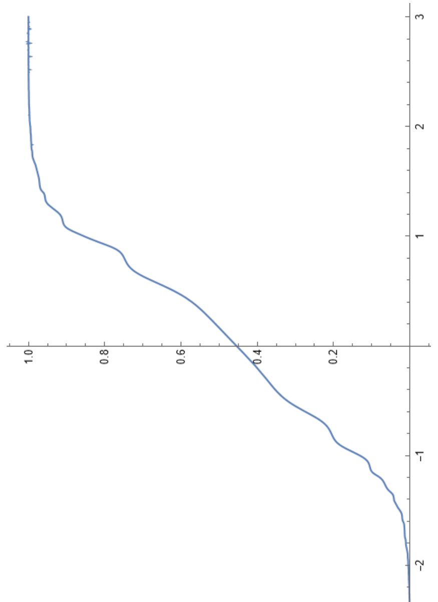
$$\Phi(x) = q \Rightarrow \frac{d\Phi(x)}{dx} dx = dq = dp$$

P(x)

$$dp = P(x)dx$$





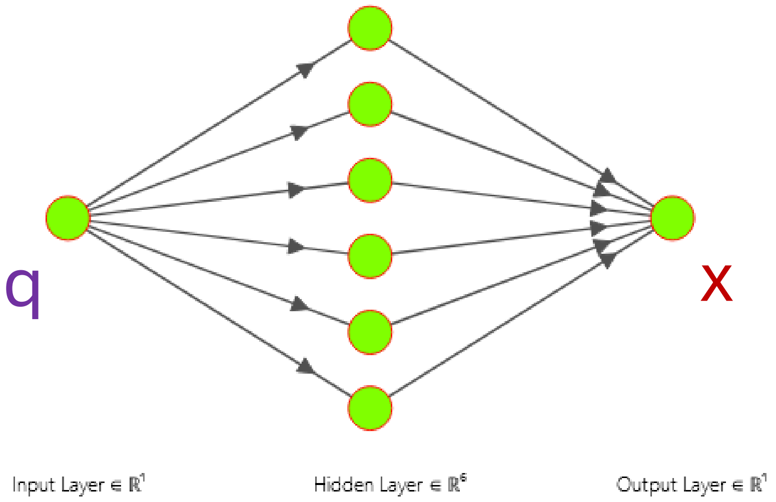


Normalizing flow

- Sample *q* from a simple probability distribution (uniform, normal...)
- Construct a mapping *q* → *x*: *x* = *F*(*q*) such that x has the required distribution

$$dx = \frac{\partial F(q)}{\partial q} dq = \frac{\partial F(q)}{\partial q} \frac{dp}{\pi(q)} \qquad \frac{dp(x)}{dx} = \pi(q(x)) \left(\left. \frac{\partial F}{\partial q} \right|_{q(x)} \right)^{-1}$$

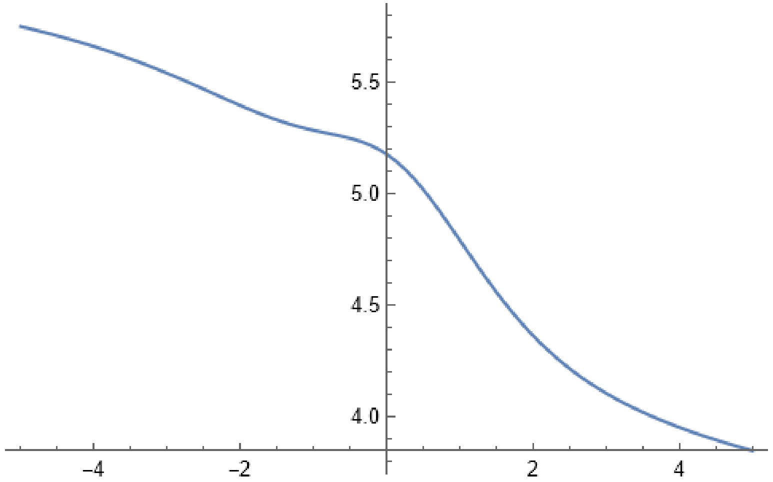
- Neural networks as universal function approximators to construct *F*(*q*)



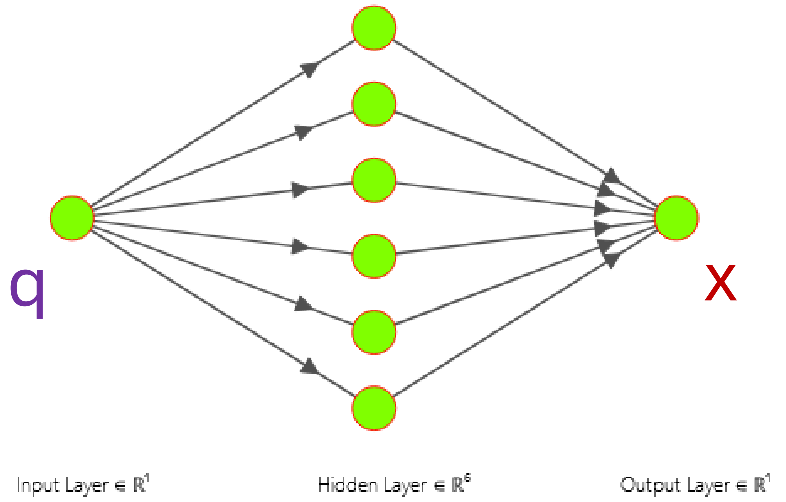
Linear transforms

Bias

$$x = \sum_i w_i^{(1)} \sigma \left(w_i^{(2)} q + b_i \right)$$



Universal function approximators

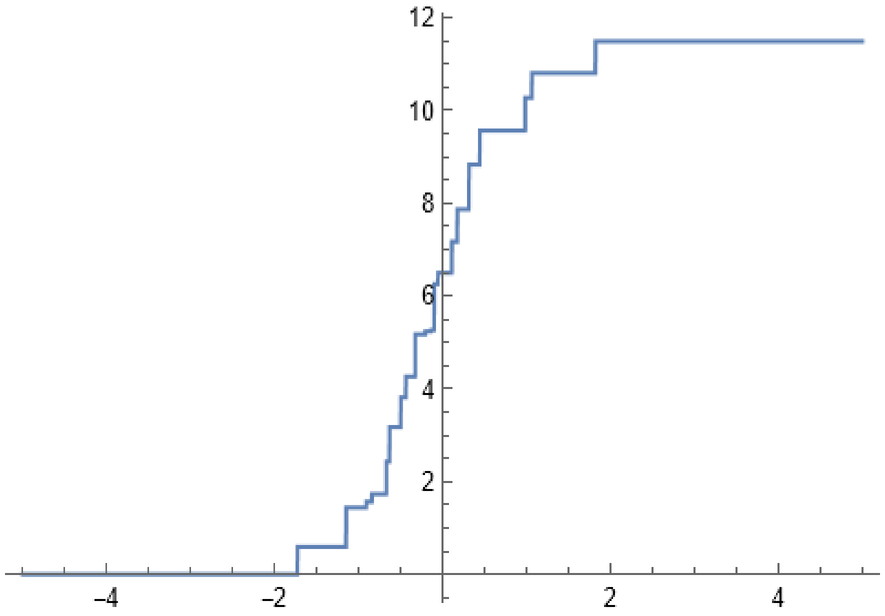


$$x = \sum_i w_i^{(1)} \sigma\left(w_i^{(2)} q + b_i\right)$$

- Let's make all $w_i^{(2)}$ very large
- rescale b_i by $w_i^{(2)}$
- Sigmoid → Heaviside step function

$$x = \sum w_i \theta\left(q - b_i\right)$$

- With positive w_i, can approximate any monotonic function
- Exactly what we need for normalizing flow



Generalizing to higher-dimensional data

- Higher-dimensional mapping $q \rightarrow x$: $x = F(q)$ such that x has the required distribution

$$d^N x = \det\left[\frac{\partial F_i(q)}{\partial q_j}\right] d^N q = \det\left[\frac{\partial F_i(q)}{\partial q_j}\right] \frac{dp}{\pi(q)}$$

$$P(x) = \left(\det\left[\left.\frac{\partial F_i(q)}{\partial q_j}\right|_{q(x)}\right]\right)^{-1} \pi(q(x))$$

Jacobian

- Higher-dimensional generalization also for universal approximation theorem
- **Avoids autocorrelation problems altogether!**
- BUT… Need to compute Jacobians for deep Neural Networks
- Computationally intensive and difficult for general NN architectures

Affine layers

$$x = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \vdots \\ \phi_n \\ \chi_1 \\ \chi_2 \\ \vdots \\ \chi_{N-n} \end{pmatrix}$$

$$\phi^i_{out} = \phi^i_{in}$$

$$\chi^k_{out} = e^{s_k(\phi_{in})}\chi^k_{in} + t_k(\phi_{in})$$

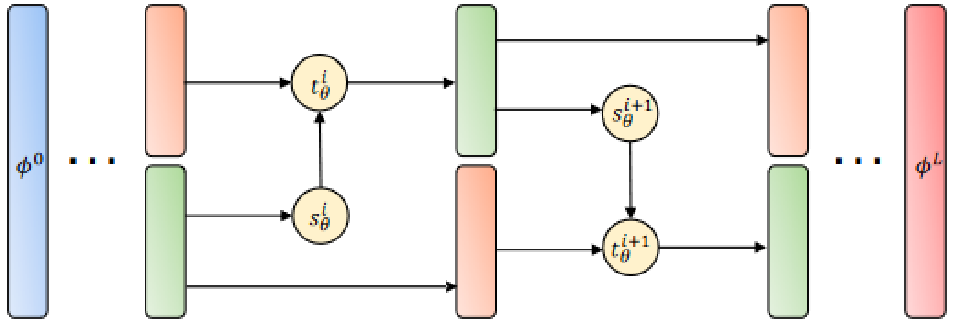
$$\frac{\partial x_{out}}{\partial x_{in}} = \left(\begin{array}{|c|c|} \hline I & \frac{\partial s_k}{\partial \phi^i_{in}} e^{s_k} \chi^k_{in} \\ \hline 0 & e^{s_k} \\ \hline \end{array} \right)$$

$$\det \begin{pmatrix} A & B \\ C & D \end{pmatrix} = \det[A] \det\left[D - CA^{-1}B\right]$$

$$\det\left(\frac{\partial x_{out}}{\partial x_{in}}\right) = \det\left(\mathrm{diag}\left(e^{s_k}\right)\right) = \prod_k e^{s_k}$$

- Jacobian is easy to compute
- s_k and t_k approximated with DNNs
- ϕ and χ are reshuffled from layer to layer
- Network still sufficiently expressive

[Diagram from ArXiv:2303.15136]



Normalizing flow – cost function

- KL divergence between *W(x) ~ exp(-S[x])* and *P(x)* :

$$\mathcal{D}_{KL}\left(P\left[x\right]|W\left[x\right]\right) = \langle \log\left(\frac{P\left[x\right]}{W\left[x\right]}\right)\rangle_{P[x]}$$

$$\mathcal{D}_{KL}\left(P\left[x\right]|W\left[x\right]\right) = \langle \sum_k s_k + S\left[x\left(q\right)\right]\rangle_{\pi[q]} + \mathrm{const}$$

- *S[x]* is the action of quantum fields *x* in (d+1)-dimensional space-time
- *x* is an abstract collective notation for discretized degrees of freedom

- Now one can use conventional stochastic optimization algorithms

- No previously generated data is necessary, as *S[x]* is known exactly!

Normalizing flow – making it exact

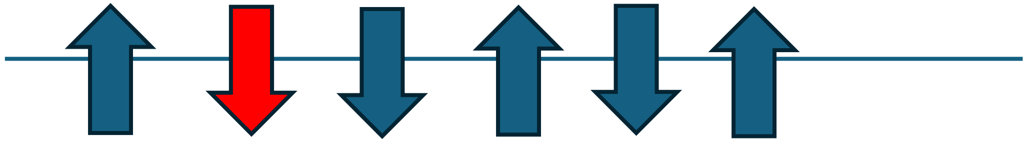
- *P(x)* is not exactly equivalent to *W(x) ~ exp(-S[x])*
- Neural nets only serve as approximation to exact normalizing flow mapping
- Use NN output as Metropolis proposal:

$$A(X|Y) = \min\left(1, \frac{W(X)}{W(Y)} \frac{P(Y|X)}{P(X|Y)}\right)$$

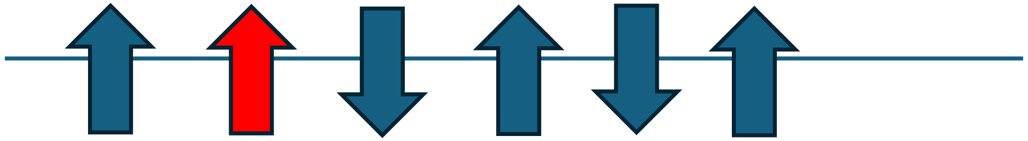
- W(X), W(Y) and P(Y|X) and P(X|Y) all computable (affine layers are invertible)
- Significantly better acceptance [ArXiv:1904.12072]
- Flow-based Markov Chain Monte-Carlo
- Applications to lattice QCD being currently developed [e.g. R. Abbot et al., ArXiv:2207.08945]
- Multi-modal distributions may still be challenging (mode collapse) [e.g. Hackett et al., 2107.00734]

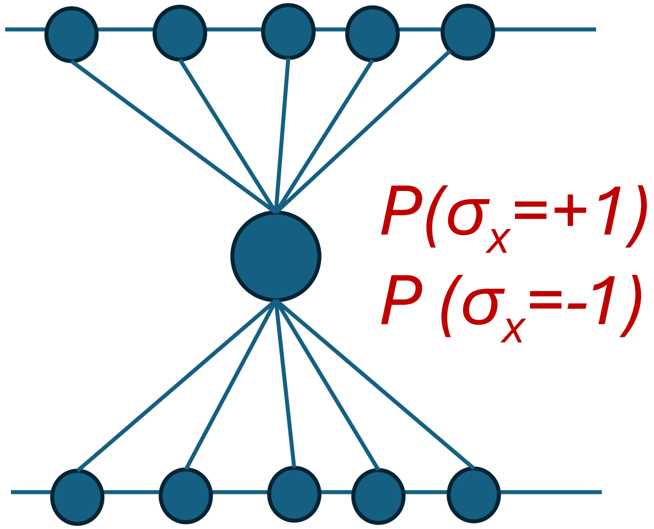
Generalizing local updates with VAEs

[ongoing work with J. Hadley]



$$P\left(\sigma_x|\sigma\right) = \mathcal{N}_x \exp\left(\beta\sigma_x\left(\sigma_{x-1}+\sigma_{x+1}\right)\right)$$





- Normalizing flow requires huge bandwidths (all degrees of freedom at once)
- Use VAEs to learn local updates: convolutions of nearest neighbours → mean and dispersion for updated values
- Produce updates rather than entire configurations → ergodic despite low dimensionality

Summary

- GenAI is a good art forger – but let's use it for good!
- Stable diffusion produces forgeries that can hardly be detected without pre-training
- Stable diffusion improves detection efficiency for human-made forgeries

- Generating random configurations of quantum fields is different from generating images/text/music…
- We have to reproduce the probability distribution exactly
- Ergodicity and dimensionality issues

- Computational cost of training? E.g. normalizing flow?