TTIC 31230, Fundamentals of Deep Learning

David McAllester, Autumn 2022

Diffusion Image Modeling Timeline

Improved Denoising Diffusion Probabilistic Models

Nichol and Dhariwal, February 2021

This paper provides a method for training an "uncertainty level" for each color channel of each pixel.

Later papers in the code base use these uncertainty levels to weight guidance strength for each color channel of each pixel in "guided diffusion".

Guided diffusion with channel-level guiding strength is used in DALLE-2.

Optimizing the ELBO

For image VAEs the ELBO is refered to as negative log likelihood (or NLL) and is measured in bits per image channel.

Model	ImageNet	CIFAR
Glow (Kingma & Dhariwal, 2018)	3.81	3.35
Flow++ (Ho et al., 2019)	3.69	3.08
PixelCNN (van den Oord et al., 2016c)	3.57	3.14
SPN (Menick & Kalchbrenner, 2018)	3.52	-
NVAE (Vahdat & Kautz, 2020)	-	2.91
Very Deep VAE (Child, 2020)	3.52	2.87
PixelSNAIL (Chen et al., 2018)	3.52	2.85
Image Transformer (Parmar et al., 2018)	3.48	2.90
Sparse Transformer (Child et al., 2019)	3.44	**2.80**
Routing Transformer (Roy et al., 2020)	**3.43**	-
DDPM (Ho et al., 2020)	3.77	3.70
DDPM (cont flow) (Song et al., 2020b)	-	2.99
Improved DDPM (ours)	**3.53**	**2.94**

Optimizing Per-Channel Prior Variances

We now introduce a prior network $\sigma_\Psi(z_\ell, \ell) \in R^d$ to give the prior noise level.

$$z_{\ell-1} = f_\Phi(\ell, z_\ell) + \sigma_\Psi(z_\ell, \ell) \odot \delta, \qquad \delta \sim \mathcal{N}(0, I)$$

The prior noise network $\sigma_\Psi(z_\ell, , \ell) \in R^d$ is trained with the ELBO objective.

This is unneccessary under the stochastic differential equation model of DDPM but improves the ELBO when using a discrete approximation to the differential equation.

Optimizing Per-Channel Prior Variances

$$z_{\ell-1} = f_\Phi(\ell, z_\ell) + \sigma_\Psi(z_\ell, \ell) \odot \delta, \quad \delta \sim \mathcal{N}(0, I)$$

Here $\sigma(z_\ell, \ell)[i]$ expresses a prior uncertainty in $z_{\ell-1}[i]$ given $z_\ell[i]$.

This uncertainty will be larger for pixels in a region with fine random texture than for pixels in a large region a contant value.

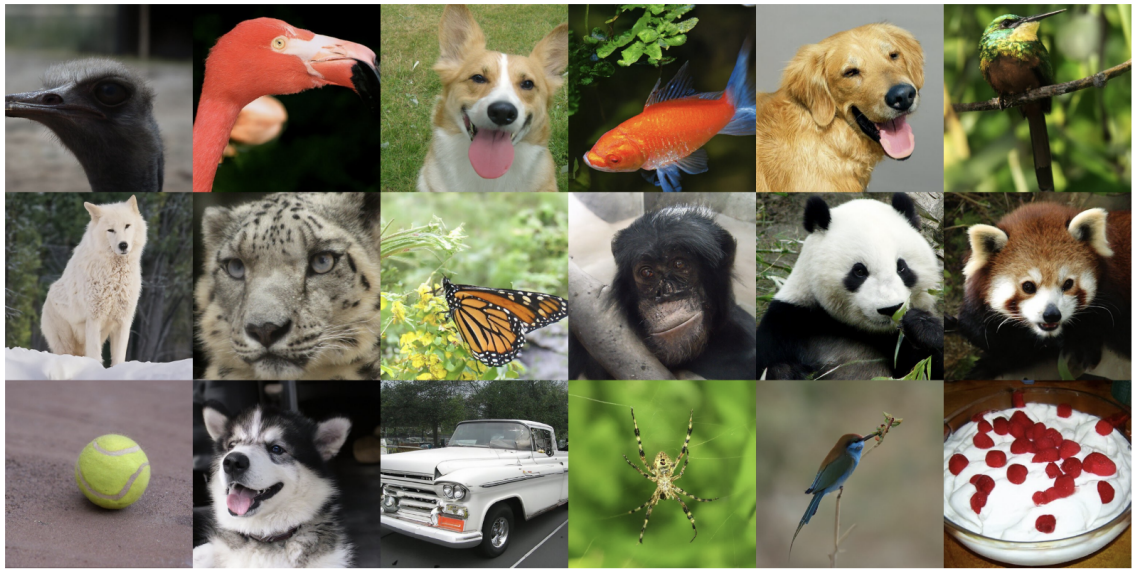
The U-Net can distinguish these different regions of the input.

The more uncertain the model at a given pixel, the more guidance should be used in adjusting it.

Diffusion Models Beat GANs on Image Synthesis

Dharwali and Nichol, May 2021

This paper introduces guided diffusion as a way of handling class-conditional diffusion models.



Guided diffusion is used in DALLE-2.

Class-Conditional Image Generation

We assume training data consisting of (x, y) pairs and we want to generate from the distribution $P(y|x)$. For example class-conditional image generation.

Previous approaches, such as StyleGAN, have trained a model (a GAN) for each class.

Here we will train a single model which takes the class label as input.

It seems that this can be made to work for VAEs but not for GANS without an auto-encoder component.

Conditional Diffusion Models

An obvious approach is to draw a pair (x, y) and pass the conditioning information x to the image genertor.

It is a weakness of GANs that we need a separate model for each x.

It seems to be a weakness of diffusion models that this natural approach to conditioning fails.

Classifier Guidance

We assume a distribution on pairs (x, y).

We also assume **a classifier** $P(x|y)$. For example x might be the ImageNET label for image y.

We will generate an image by using $P(x|y)$ to "guide" generation from the unconditional model $\epsilon(z_\ell, \ell)$.

Guidance will be based on the score-mathching interpretation of diffusion models where $\epsilon(z_\ell, \ell)$ is interpreted as $-\nabla_z \ln p(z)$.

Class-Conditional Generation

$$z_{\ell-1} \;=\; \frac{1}{\sqrt{1-\sigma_\ell^2}}\,(z_\ell - \sigma_\ell\;\epsilon(z_\ell,\ell)) \;+\; \tilde{\sigma}_\ell \odot \delta$$

Score-matching interprets $\epsilon(z_\ell, \ell)$ as $-\nabla_z \ln p(z)$.

$$p(z|x) = \frac{P(z)P(x|z)}{P(x)} \propto p(z)P(x|z)$$

We want a step in direction $\nabla_z \;\ln p(z)P(x|z)$. They use

$$\mathrm{pri}(z_\ell,\ell) \;=\; \frac{1}{\sqrt{1-\sigma_\ell^2}}\,(z_\ell - \sigma_\ell\;\epsilon(z_\ell,\ell)) \;+\; \tilde{\sigma}_\ell \odot \delta \;+ s\tilde{\sigma} \odot \nabla_z \ln P(x|z)$$

Classifier Guidance

$$\text{pri}(z_\ell, \ell) \;=\; \frac{1}{\sqrt{1-\sigma_\ell^2}}\,(z_\ell - \sigma_\ell\,\epsilon(z_\ell, \ell)) \;+\; \tilde{\sigma}_\ell \odot \delta \;+ s\tilde{\sigma} \odot \nabla_z \ln P(x|z)$$

Here s is called the scale of the guidance.

Empirically it was found that $s > 1$ is needed to get good class-specificity of the generated image.

However, increasing s decreases diversity so we have a diversity/quality trade off.

Other Improvements

Various architectural choices in the U-Net were optimized based on FID score (not NLL).

These improvements are used in DALLE-2.

Classifier-Free Diffusion Guidance

Ho and Salimans, December 2021 (NeurIPS workshop)

Classification diffusion guidance uses a classifion model $P(x|y)$.

This paper introduces "classifier-free" diffusion guidance.

Classifier-free diffusion guidance is used in DALLE-2.

Classifier-Free Diffusion Guidance

5% of the time we set $x = \emptyset$ where $\emptyset$ is a fixed value unrelated to the image.

The prior then uses

$$\tilde{\epsilon}(z_\ell, \ell, x) = s\epsilon(z_\ell, \ell, x) - (s-1)\epsilon(z_\ell, \ell, \emptyset)$$

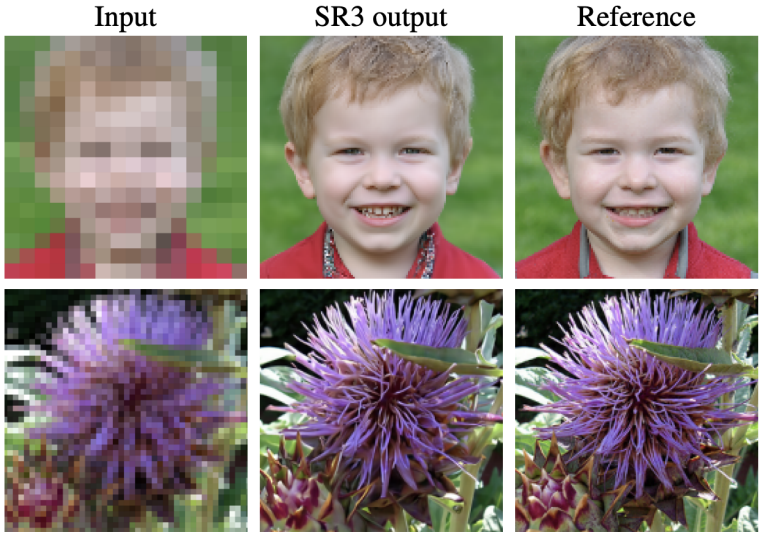
where $s \geq 1$ controls the relative weight of the two terms.

DALLE-2 incorporates the channel-level uncertainties $\tilde{\sigma}$ as weights on classifier-free diffusion guidance provided by CLIP.

Image Super-Resolution via Iterative Refinement

Saharia, Ho et al., April 2021

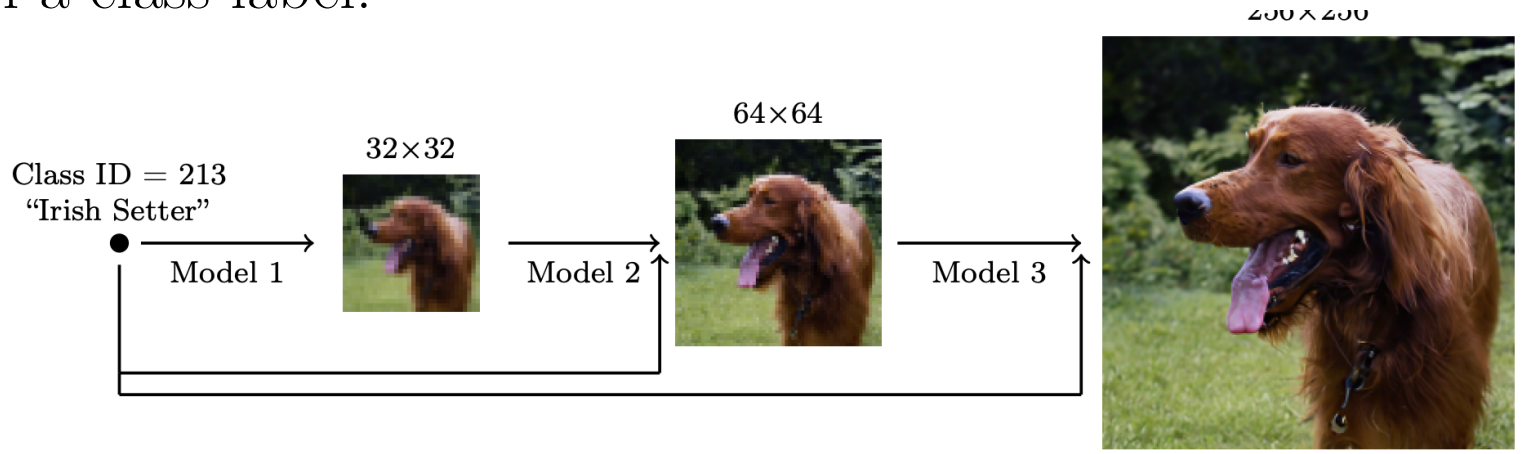
They construct a super-resolution diffusion model as conditional model for pairs for pairs (x, y) with x is a downsampling of y.



Cascaded Diffusion Models ...

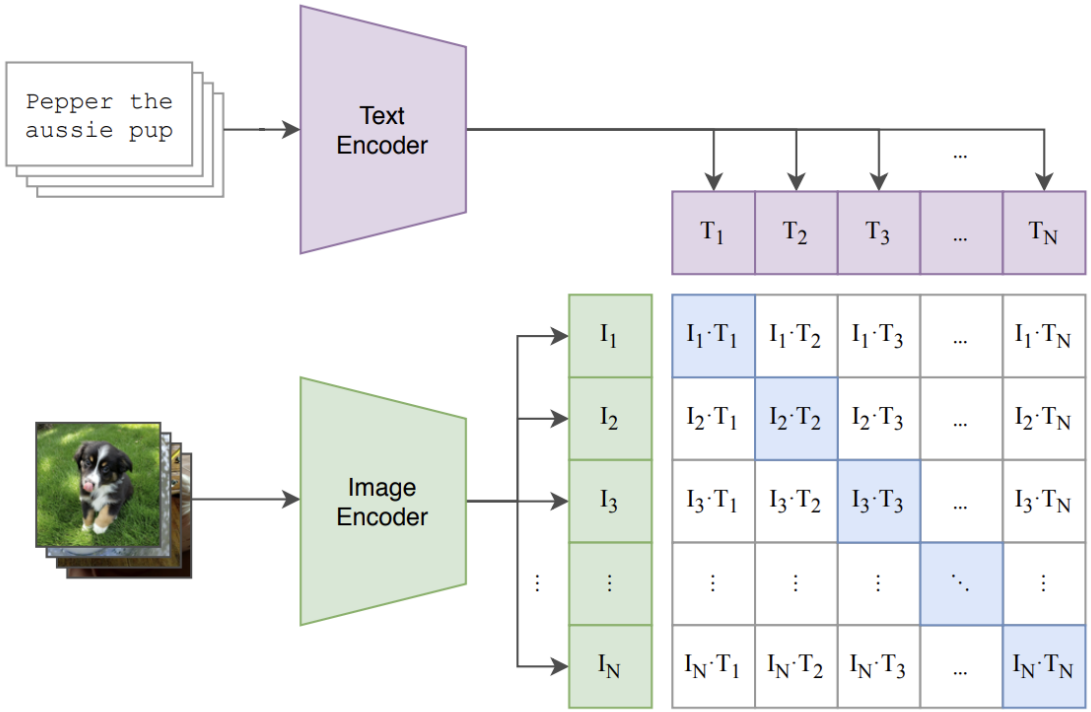
Ho, Saharia et al, May 2021

A series of super-resolution diffusion models each conditioned on a class label.



This architecture is used in DALLE-2.

CLIP Does Contrastive Coding



CLIP is used in DALLE-2 and in DALLE-2's predicessor GLIDE.

GLIDE: Towards Photorealistic Image Generation ...

Nichol, Dhariwal, Ramesh, et al., December 2021

GLIDE compares two forms of diffusion guidance.

(a) Classifier-free guidance based on comparing conditioned and unconditioned decoding directions.

(b) Classifer guidance based on CLIP.

Classifier-free (self-guided) GLIDE

$$\tilde{\epsilon}(z_\ell, \ell, x) = s\epsilon(z_\ell, \ell, x) - (s-1)\epsilon(z_\ell, \ell, \emptyset)$$

Classifier-free GLIDE does not use CLIP.

The classifier-free guidance differs from the original version in that here we are conditioning on text rather than as Imagenet labels.

The text is transformed to a feature vector by a transformer before being fed to the prior.

CLIP-guided GLIDE

Let $C_I(y)$ be the CLIP vector for image y and let $C_T(x)$ be the CLIP vector for text x.

$$z_{\ell-1} = \frac{1}{\sqrt{1-\sigma_\ell^2}}\left(z_\ell - \sigma_\ell\,\epsilon(z_\ell,\ell) + s\tilde{\sigma}\odot\nabla_z C_T(x)^\top C_I(z)\right) + \tilde{\sigma}_\ell\odot\delta$$

Here CLIP is re-trained to handle noised images.

Upsamling

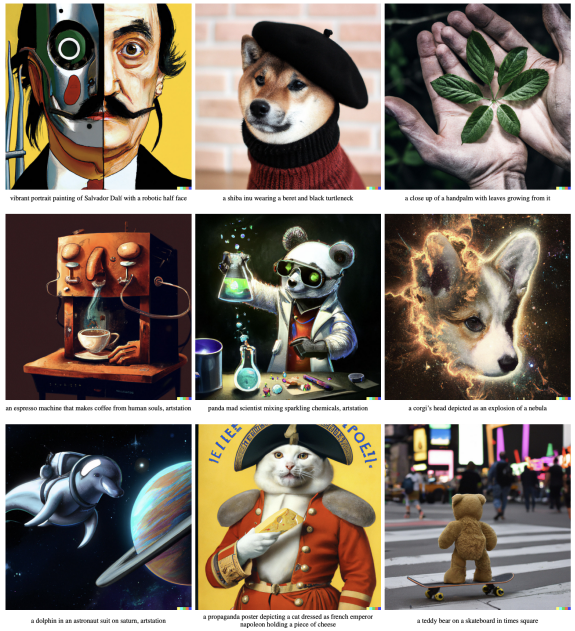
Both GLIDE versions use diffusion upsampling to go from 64×64 to 256×256.

The GLIDE paper concludes that the classifer-free model taking raw text as input is superior to the CLIP-guided model.

DALL·E-2

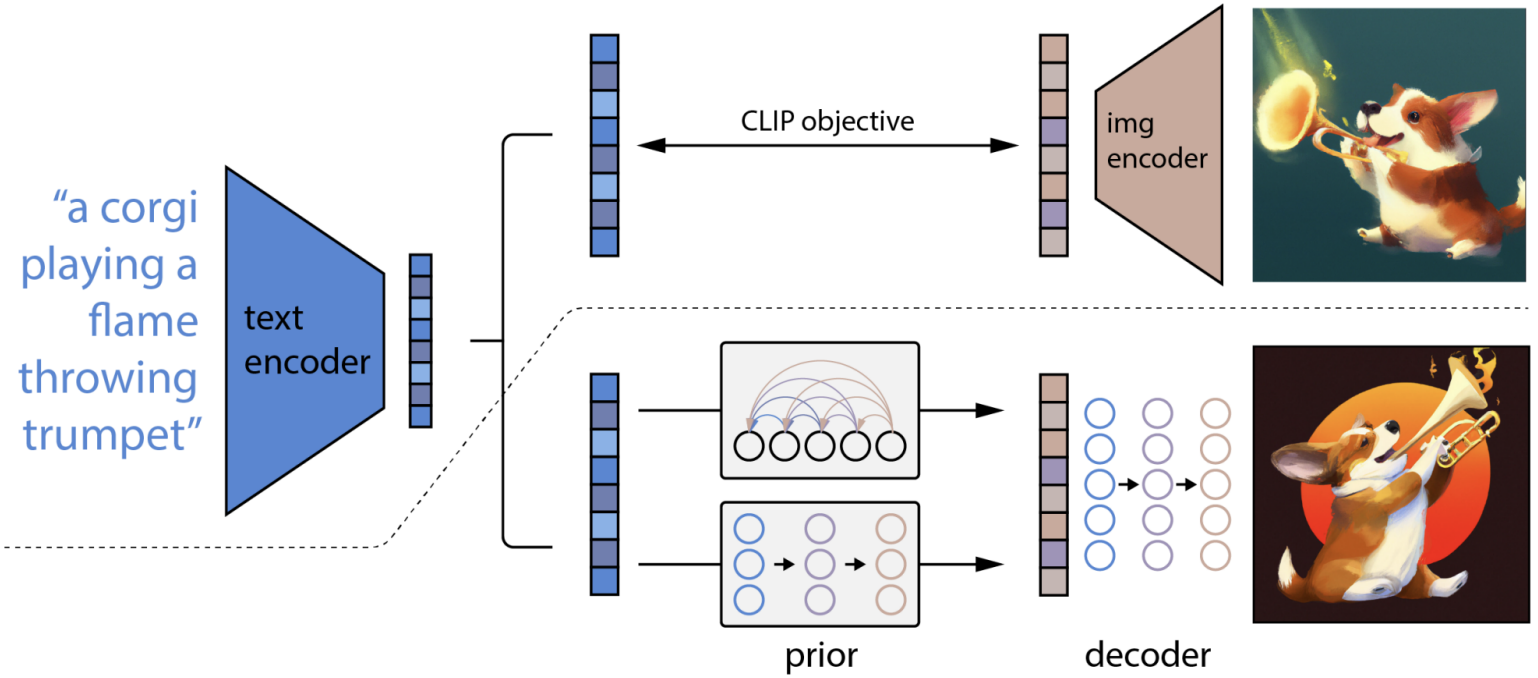
Ramesh, Nichol, Dhariwal, et al., March 2022

panda mad scientist mixing sparkling chemicals, artstation



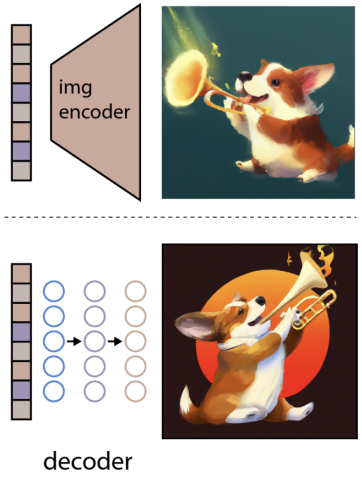
CLIP-guided DALLE-2 is similar in quality to self-guided GLIDE but is more diverse.

DALL·E-2



This figure is misleaning. The lines in the figure do not correspond to the actual data paths of DALLE-2.

A Conditional Image Auto-Encoder



Let $C_I(y)$ denote the CLIP embedding of image y.

$C_I(y)$ is the encoder of an auto-encoder for y given x.

$P(C_I(y)|x)$ is the optimal prior for this auto-encoder.

$P(y|C_I(y), x)$ is the optimal prior.

In DALLE-2 the prior and the prior both see the text x.

Putting it all Together

We are given text x.

Draw $\hat{C}$ from the prior $P(C_I(y)|x)$

Do diffusion decoding with two upsampling models:

$$\text{compute } \tilde{z}_{\ell-1} \text{ using } \hat{\epsilon} = s\epsilon(z_\ell, \ell, x) - (s-1)\epsilon(z_\ell, \ell, \emptyset)$$

$$z_{\ell-1} = \hat{z}_{\ell-1} + s'\tilde{\sigma} \odot \nabla_z \hat{C}^\top C_I(z)$$

The Prior

They experiment with two priors $P(C_I(y)|x)$.

An autoregressive model and a conditional diffusion model.

They say both priors use self-guidance.

The Autoregressive Prior

First do PCA on the distibution of vectors $C_I(y)$ to reduce their dimensionality from 1024 to 319.

Sort the eigenvectors in decreasing order of eigenvalue.

Quantize each of 319 values into 1024 discrete buckets.

We train a transformer to take the text sequence x followed by the text embedding $C_T(x)$ and to and to predict a string of 319 symbols with a vocabulary of size 1024 which can be converted back into the vector $\hat{C}$.

The Diffusion Prior

Let z_ℓ be the noising of $C_I(y)$ to level ℓ. For the prior they train a transformer to take the text string x, the text embedding $C_T(x)$, the noised image embedding z_ℓ, and the level ℓ. A final "classifier token" is added to the end of this string and vector computed for that token by the transformer is used as a prediction of $C_I(y)$. This predictor $f(x, z_\ell, \ell)$ is trained on the objective

$$f^* = \underset{f}{\text{argmin}} \; E_{x,y,z_\ell,\ell} \, ||f(x, z_\ell, \ell) - C_I(y)||^2$$

Sampling $\hat{C}$ from the Diffusion Prior

The paper does not describe the decoding process that computes $z_{\ell-1}$ from z_ℓ but the following seems reasonable.

$$z_{\ell-1} = f(x, z_\ell, \ell) + \tilde{\sigma}\delta \quad \delta \sim \mathcal{N}(0, I)$$

They can draw samples of $\hat{C}$ using different values δ

$$z_0 = f(x, z_1, 1) + \tilde{\sigma}\delta \quad \delta \sim \mathcal{N}(0, I)$$

They draw two samples $\hat{C}$ and $\hat{C}'$ and use the one with larger inner product with $C_T(x)$.

Markovian VAEs

Diffusion models are a special case of Markovian VAEs.

A Markovian VAE has latent variable $z = (z_0, z_1, \ldots, z_L)$.

It is not clear whether diffusion models are the best way to do this.

It could be that the important idea is simply having large L with $I(z_0, z_\ell)$ going from $H(z_0)$ to 0 as ℓ goes from 0 to L.

END