Constructions for Diffusion that Leverage HAE Guarantees
CS6360: Final Report

Pranav K Nayak *es20btech11035@iith.ac.in*
Department of Engineering Science
IIT Hyderabad

Abstract

The Homomorphism Autoencoder (HAE) provides theoretical guarantees for the learned representations of transformations of its input. This report details my attempt at ensuring that the process of diffusion fits the assumptions of all the theorems of the HAE, thereby allowing it to learn representations for diffusion that have the same structural guarantees.

1 The Homomorphism Autoencoder

The Homomorphism Autoencoder, from Keurti et al. (2023), is an autoencoder framework that learns not just a representation of the input, but also representations of *transformations* of the input. These representations have certain structural guarantees, formalized using the language of groups and smooth manifolds. These structural properties are only guaranteed, however, if certain assumptions about the data-generating process are taken to be true.

1.1 The HAE Architecture

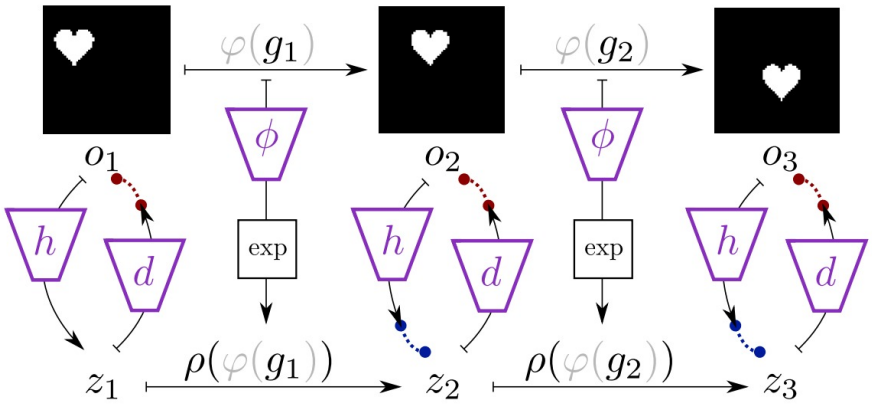

Figure 1: The HAE's architecture (figure borrowed from Keurti et al. (2023))

Figure 1 details the architecture of the autoencoder. It consists of three deterministic neural networks:

- the encoder h, that learns representations of the inputs (henceforth called the *observations*)
- the decoder d, a decoder for h, and
- the transformation encoder ϕ, which, if given a transformation in the space of observations, learns an analogous one in the space of latents.

The input to the model consists of a trace of observations and actions. The model is trained to minimize a combined prediction-reconstruction loss, the details of which can be found in section 1.3.

1.2 A Walkthrough of the Encoding Process

For the purposes of illustration, let us assume that a single transformation has occurred, meaning that the trace is of the form (o_1, g_1, o_2). The latents computed by h for both observations are given as

$$z_1 = h(o_1) \quad z_2 = h(o_2).$$

The representation of the transformation in latent space is given by [1]

$$\tilde{g}_1 = \phi(g_1).$$

There are two "gaps" in the encoding process that are the targets of minimization:

1. The *prediction* gap $||z_2 - \tilde{g}_1(z_1)||_2^2$, and
2. The *reconstruction* gap $||o_2 - d(\tilde{g}_1(z_1))||_2^2$.

Clearly, minimizing the prediction gap works to ensure that transformations in the latent space are consistent with transformations in the observation space, and minimizing the reconstruction gap works to ensure that the observation encoder actually learns something meaningful. Minimizing the reconstruction gap also plays the critical role of ensuring that ϕ does not end up being the trivial (identity) representation.

1.3 Training Loss

For a trace $(o_1, g_1, o_2, g_2, ..., g_{N-1}, o_N)$, the full form of the losses are given by:

$$L^N_{\text{pred}} = \sum_{t=2}^{N+1} ||z_t - \big(\prod_{i=1}^{t-1} \phi(g_i)\big)h(o_1)||_2^2, \text{ and}$$

$$L^N_{\text{rec}} = \sum_{t=2}^{N+1} ||o_t - d\Big(\prod_{i=1}^{t-1} \phi(g_i)h(o_1)\Big)||_2^2.$$

Let us analyze the prediction loss. For any single term of the summation, $||z_t - \big(\prod_{i=1}^{t-1} \phi(g_i)\big)h(o_1)||$, the minimizer is that h and ϕ result in the *cumulative prediction gap* getting minimized. The cumulative minimization gap is the gap between the encoding of the t^{th} observation, and the predicted encoding when we start from the very first latent z_1.

Thus, minimzing the sum is equivalent to ensuring that no matter how many transformations are performed on the observations, we need only take the first encoding, and operate exclusively in latent space, composing the transformations $\phi(g_i)$ to get the encoding of the final observation.

Similar intuition can be built for the reconstruction loss. It essentially minimizes the same gap that the prediction loss does, but projected back into observation space.

The combined loss is given by $L^N_{\text{pred}} + \gamma L^N_{\text{rec}}$.

1.4 Structural properties of HAE representations

Representations of observations: The encoder $h : O \to Z$ learns *group-structured representations* (Higgins et al., 2018) if

- Z permits the group acting on O to act on itself.
- The encoder h is equivariant to the group action, i.e., for all $o \in O$ and $g \in G$, $h(g_O o) = g_Z h(o)$.

[1] This is not entirely true. The details of the interaction between ϕ, φ, and exp can be found in appendix A.1.

Representations of transformations: The representation ϕ is *disentangled* with respect to some decomposition $(G_1 \times G_2 \times ... \times G_n)$ of G if

- Z is also decomposable into the same number of units n as G is.
- ϕ is also decomposable into n units $\phi_1, \phi_2, ..., \phi_n$ (this is enforced through a sparsity regularizer)
- Each unit Z_i permits the group G_i to act on it.
- The action of G on Z (the form of which is learned by ϕ) is decomposable as

$$g_Z z = \phi(g_1 \times g_2 \times ... \times g_n)(z_1 \times z_2 \times ... \times z_n) = \phi_1(g_1)z_1 \times \phi_2(g_2)z_2 \times ... \times \phi_n(g_n)z_n.$$

There are multiple justifications for why such properties are desirable: that they allow for improved interpretability; that control over representations is easier; that they align better with physics-based real-world transformations. Higgins et al. (2018) and Higgins et al. (2022) give significantly more detailed arguments in favour of these properties than will be presented here.

1.5 Assumptions about the Data-Generating Process

Below, I expand on the assumptions needed for the structure of section 1.4 to hold. These assumptions are taken from Keurti et al. (2023).

The process by which the data are generated involves a smooth manifold W of world states, and a vector space $W*$ on which transformations are performed. It is assumed that there is a diffeomorphic map $m : W \to W*$. The action of $g \in G$ on W is actually taken to be the result of mapping into $W*$, taking the action of g in $W*$, and then mapping the result back into W:

$$g_W w \triangleq m^{-1}(g_{W*} m(w)) \text{ where } w \in W$$

The world states are mapped to observations by a diffeomorphic map $b : W \to O$, and actions of a group on the observation space can actually be arrived at through actions of the group on $W*$. Figure 2 should help build intuition about the relationship between the different sets.

If these assumptions are adhered to, then the properties of section 1.4 hold, in accordance with proposition 1 of Keurti et al. (2023).

Thus, we arrive at the main idea behind this report:

1.6 Central Motivation

If, given a set of transformations G, we can construct set W, space $W*$, mappings m and b, and represent the transformations as elements of some matrix group G_{W*} acting on $W*$, then we have that the HAE-representations for the set G are symmetry-based and disentangled.

2 The Diffusion Process

I have selected *diffusion*, as detailed in Sohl-Dickstein et al. (2015), as the set of transformations for which the constructions of 1.6 will be attempted. The motivations behind this are simple:

- The diffusion process, at a first glance, fits the framework of iterative observations and transformations that the HAE is designed for.
- Symmetry-based and disentangled representations of the diffusion process could potentially be interpretable, and at the very least provide control during generation.
- The learned representations of the denoising process might have performance benefits, since the HAE is designed to learn representations that are useful for prediction and reconstruction.

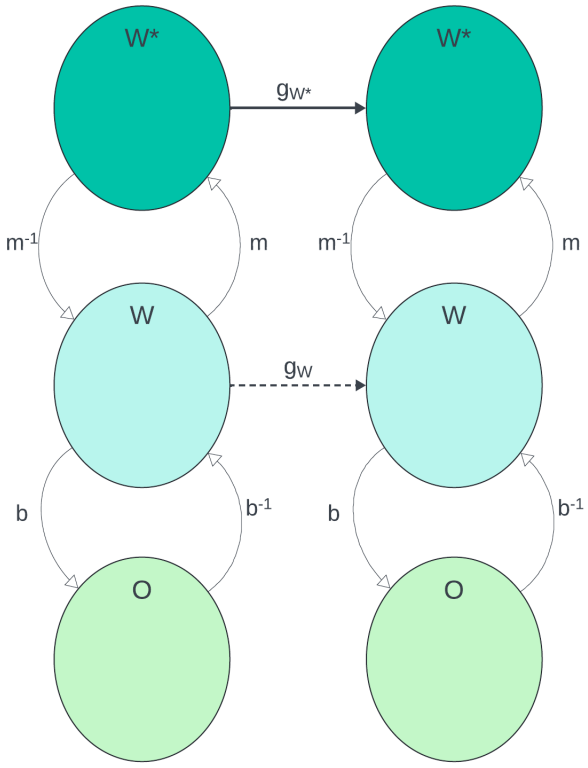

Figure 2: An illustration of the sets and spaces involved in the data-generating process

- The learned representations could be interesting in their own right, potentially shedding light on notions of geometry and symmetry in the diffusion process, similar to the results of Keurti et al. (2023).

While most of these benefits can only be verified through empirical study, showing that the constructions are possible is the first step, indicating that experimentation is warranted. I will now detail the requisite constructions.

2.1 The Set W

Since our observation space is pixel space, i.e. the set of all $n \times n$ matrices, I construct W to be the space R^{n^2}, with b^{-1} being the linearization operation, and b the reconstruction operation. It is required to prove that b is a *diffeomorphism*, which can be done by proving the following:

- W and O are smooth manifolds.
- There exist projections from both W and O onto some finite-dimensional vector space, such that the map between these projections that is equivalent to b is smooth, invertible, and whose inverse is smooth.

From example 1.24 of Lee (2013), we have that since both W and O are smooth manifolds, since they are both finite-dimensional vector spaces. If we take the projection from W onto $\mathbb{R}^{n^2}$ to be the linearization (which is the same as taking $\psi = b$), and taking the projection from O onto $\mathbb{R}^{n^2}$ to be the identity, then the map between these projections is the identity, which is smooth, invertible, and whose inverse is smooth. Thus, b is a diffeomorphism.

2.2 The Space $W*$

I construct $W*$ to be the space $\mathbb{R}^{n^2+1}$. The map $m : \mathbb{R}^{n^2} \to \mathbb{R}^{n^2+1}$ is defined as

$$m([v_1, v_2, ..., v_{n^2}]^T) = [v_1, v_2, ..., v_{n^2}, 1]^T$$

Note that m is not a surjection, and thus can only be a bijection if we restrict $W*$ to the image of W under m. This would, however, remove its status as a vector space. Therefore, we must accept the slightly weakened construction of $W* = \text{Im}_m(W) + \mathbf{0}_{n^2+1}$. Since in practice, we will never encounter the vector $\mathbf{0}_{n^2+1}$, it is of practical relevance to proceed with $m^{-1} : \text{Im}_m(W) \to W$ as the inverse we would have gotten otherwise:

$$m^{-1}([v_1, v_2, ..., v_{n^2}, 1]^T) = [v_1, v_2, ..., v_{n^2}]^T.$$

From similar reasoning as in the previous section, m is a diffeomorphism. (Here, the finite-dimensional vector space that both W and $W*$ are mapped onto is $\mathbb{R}^{n^2}$)

2.3 The Matrix Group G_{W*}

In order to represent the denoising operation as arising from a matrix multiplication (and therefore as the result of a group's action on a set), we must first relate the structure of the denoising process to that of the noising process. This is due to the fact that the noising process, being pre-determined and Gaussian, can be written down as

$$x_{k+1} = x_k + \sigma\epsilon \text{ where } \epsilon \sim \mathcal{N}(0, I).$$

If the x_i's belong to $\mathbb{R}^n$, then the above operation can be represented as a matrix multiplication in $\mathbb{R}^{n+1}$:

$$x_{k+1} = \left(\begin{bmatrix} 1 & 0 & 0 & ... & 0 & \sigma\epsilon_1 \\ 0 & 1 & 0 & ... & 0 & \sigma\epsilon_2 \\ 0 & 0 & 1 & ... & 0 & \sigma\epsilon_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & ... & 1 & \sigma\epsilon_n \\ 0 & 0 & 0 & ... & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k,1} \\ x_{k,2} \\ x_{k,3} \\ \vdots \\ x_{k,n} \\ 1 \end{bmatrix} \right)_{[1:n]}.$$

Thus, if a similar update equation can be arrived at for the denoising process, then we can represent it too as a matrix-multiplication.

To do so, I refer to equation 58 of Luo (2022), which offers a lower-bound on the ELBO:

$$\log p(x) \geq \underbrace{\mathbb{E}_q(x_1|x_0)[\log p_\theta(x_0|x_1)]}_{\text{reconstruction term}} - \underbrace{D_{KL}(q(x_T|x_0)||p(x_T))}_{\text{prior matching term}} - \sum_{t=2}^{T} \mathbb{E}_q(x_t|x_0) \underbrace{[D_{KL}(q(x_{t-1}|x_t, x_0)||p_\theta(x_{t-1}|x_t))]}_{\text{denoising matching term}}.$$

The *denoising matching term* tells us that it makes sense to have the learned denoiser $p_\theta(x_{t-1}|x_t)$ be as close as possible to the ground-truth denoiser $q(x_t|x_{t-1}, x_0)$.

From equation 84 of Luo (2022), we have that the denoising process is distributed as

$$q(x_t|x_{t-1}, x_0) \propto \mathcal{N}(\underbrace{\frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})x_t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)x_0}{1-\bar{\alpha}_t}}_{\mu_q(x_t,x_0)}, \underbrace{\frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} I}_{\Sigma_q(t)}).$$

Thus, we have that the ground-truth denoising process is also normally distributed, meaning that the learned one can be parameterized as $\mathcal{N}(\mu_\theta(x_t), \Sigma(t))$, with the mean being the output of a neural network, seeing as we do not have access to the ground-truth image x_0.

Thus, the denoising update rule for a learned denoising process p_θ can be written as

$$x_{t-1} = \mu_\theta(x_t) + \Sigma(t)\epsilon \text{ where } \epsilon \sim \mathcal{N}(0, I),$$

which can, in turn, be represented as a matrix multiplication in $\mathbb{R}^{n+1}$:

$$\begin{bmatrix} x_{t-1,1} \\ x_{t-1,2} \\ x_{t-1,3} \\ \vdots \\ x_{t-1,n} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{\mu_\theta(x_t)_1}{x_{t,1}} & 0 & 0 & ... & 0 & \Sigma(t)\epsilon_1 \\ 0 & \frac{\mu_\theta(x_t)_2}{x_{t,2}} & 0 & ... & 0 & \Sigma(t)\epsilon_2 \\ 0 & 0 & \frac{\mu_\theta(x_t)_3}{x_{t,3}} & ... & 0 & \Sigma(t)\epsilon_3 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & ... & \frac{\mu_\theta(x_t)_n}{x_{t,n}} & \Sigma(t)\epsilon_n \\ 0 & 0 & 0 & ... & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{t,1} \\ x_{t,2} \\ x_{t,3} \\ \vdots \\ x_{t,n} \\ 1 \end{bmatrix}.$$

Thus, we have that denoising can be represented as a matrix-multiplication. To ensure that the set that these matrices come from is a (compact) group, we just set it to be the subset of $GL(n^2+1, \mathbb{R})$ whose elements' absolute values do not exceed a constant $C \sim 1$ (assuming the dynamic range of images is between 0 and 1). Compactness here arises from the fact that both $x_{t,i}$ and $\mu_\theta(x_t)_i$ represent pixel values, and this means that each element is bounded.[2].

3 Weaknesses

The main weakness of this approach appears to be that the final column of the matrix group is always the result of a stochastic process. While we could simply hand-wave away the stochasticity by attributing it to the randomness of sampling from the set of all possible transformations, the fact that the individual elements are Gaussian suggests that something more concrete can be said about this form of the diffusion group.

An intuitive direction of inquiry is to see if any of my assertions can be said to hold in expectation. While this is trivial over a single step (since the last column would all get clipped down to zero, leaving us with a linear transformation), the fact that the mean changes at every step means that the distribution over the entire denoising process is significantly more complex.

References

B. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction.* Graduate Texts in Mathematics. Springer International Publishing, 2015. ISBN 9783319134673.

Brian C. Hall. An elementary introduction to groups and representations, 2000.

Irina Higgins, David Amos, David Pfau, Sebastien Racaniere, Loic Matthey, Danilo Rezende, and Alexander Lerchner. Towards a definition of disentangled representations, 2018.

Irina Higgins, Sébastien Racanière, and Danilo Rezende. Symmetry-based representations for artificial and biological general intelligence, 2022.

Hamza Keurti, Hsiao-Ru Pan, Michel Besserve, Benjamin F. Grewe, and Bernhard Schölkopf. Homomorphism autoencoder – learning group structured representations from observed transitions, 2023.

J.M. Lee. *Introduction to Smooth Manifolds.* Graduate Texts in Mathematics. Springer New York, 2013. ISBN 9780387217529.

Calvin Luo. Understanding diffusion models: A unified perspective, 2022.

Jascha Sohl-Dickstein, Eric A. Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics, 2015.

[2]I am not certain of the exact form of the bound, but it clearly depends on $\Sigma(t)$, and therefore on the step-size scheme α_t.

A Appendix

A.1 A Note on Lie Algebras and Lie Groups

If we are to work with groups, then there is not necessarily any restriction on what form the group's elements can take. They need only satisfy the group axioms, which means that any set of arbitrarily complex structures that is composable, closed under composition, and contains an identity and inverses, can be used to represent the set of transformations we are interested in. If, however, the set of transformations can be represented as elements of a smooth manifold, then there are certain simplifications that can be made.

A group that is also a smooth manifold is termed a *Lie group*. These manifolds are not necessarily vector spaces, since while they might belong to some subset $E \in \mathbb{R}^n$, we do not require that $\mathbf{0} \in E$. It is helpful to think of them as subsets that are offset from the origin by some amount.

Given some Lie group G, the tangent space of this manifold around the identity element forms its *Lie algebra* $\mathfrak{g}$. The Lie algebra has the benefit of being a vector space. We cannot immediately use the algebra to represent the group, however, since it is a linear approximation of the group at a point, and thus incurs some error.

Before we begin to tackle this error, we must first understand the notion of the *representation* of a group on a set. Intuitively, the group's representation on a set is the form that the elements of the group take to be able to act on the set. For a concrete example, consider the group G of invertible linear transformations. Without specifying exactly what space/set is being transformed, the group remains as an abstract concept. If, however, we specify that we are working with $\mathbb{R}^n$, we can say that the representation of G on this space is the set of invertible $n \times n$ matrices, which just so happens to have the notation $GL(n, \mathbb{R})$.

Since we are working with concrete, real vector spaces, it helps to speak of groups in terms of their representations on these spaces. From now on, let us treat the representation of any group on the vector space we are working on as the group itself, and call it the *matrix* group for that space.

From definiton 2.2 of Hall (2000), we have that any *closed subgroup* of $GL(n, \mathbb{R})$ is a Lie group. Corollary 3.44 of Hall (2015) tells us that if our lie group is *compact*, then there is a bijective map between the group and its lie algebra.

This bijective map is given by $\exp : \mathfrak{g} \to G$, defined as

$$e^X = \sum_{m=0}^{\infty} \frac{X^m}{m!}$$

From definition 2.4 of Hall (2015), a matrix group is compact if $\exists C \in \mathbb{R}$ such that for all $A \in G$, $|A_{ij}| \leq C$ for all $1 \leq i, j \leq n$.

From proposition 2.1 of Hall (2015), the matrix exponential converges for all square X.

Thus, in summary, we have that if we are working with a compact matrix group, we can instead work with its lie algebra, with the knowledge that the two are equivalent. This realization manifests in the HAE through the fact that ϕ actually takes elements of the lie algebra as input.

Let the matrix group acting on observation space be denoted by G, with a lie algebra of $\mathfrak{g}$, and the matrix group acting on the latent space be denoted by $\tilde{G}$, with a lie algebra of $\tilde{\mathfrak{g}}$. Then ϕ learns to map between $\mathfrak{g}$ and $\tilde{\mathfrak{g}}$, after which the exponential map $\exp : \tilde{\mathfrak{g}} \to \tilde{G}$ is used to get a valid matrix transformation for the latent space.

A.2 A Note on Diffeomorphisms

A central idea for proving smoothness is the notion of a *coordinate chart*. A coordinate chart (or just a chart) on a manifold M is a pair (U, ϕ), where U is an open subset of M, and $\phi : U \to \mathbb{R}^n$ is a homeomorphism. The homeomorphism property is what allows us to map between the manifold and $\mathbb{R}^n$. In order to prove that a function f between two manifolds M and N is smooth, we need to show that for point $p \in M$, we

can find charts (U, ϕ) and (V, ψ) such that $p \in U$, $f(p) \in V$, and the function f, written as $\psi \circ f \circ \phi^{-1}$, is a smooth function between $\mathbb{R}^n$ and $\mathbb{R}^n$.

For a more complete treatment of the smooth manifolds, see Lee (2013).