CFASL: Composite Factor-Aligned Symmetry Learning for Disentanglement in Variational AutoEncoder

Anonymous authors
Paper under double-blind review

Abstract

Symmetries of input and latent vectors have provided valuable insights for disentanglement learning in VAEs. However, only a few works were proposed as an unsupervised method, and even these works require known factor information in training data. We propose a novel method, Composite Factor-Aligned Symmetry Learning (CFASL), which is integrated into VAEs for learning symmetry-based disentanglement in unsupervised learning without any knowledge of the dataset factor information. CFASL incorporates three novel features for learning symmetry-based disentanglement: 1) Injecting inductive bias to align latent vector dimensions to factor-aligned symmetries within an explicit learnable symmetry codebook 2) Learning a composite symmetry to express unknown factors change between two random samples by learning factor-aligned symmetries within the codebook 3) Inducing group equivariant encoder and decoder in training VAEs with the two conditions. In addition, we propose an extended evaluation metric for multi-factor changes in comparison to disentanglement evaluation in VAEs. In quantitative and in-depth qualitative analysis, CFASL demonstrates a significant improvement of disentanglement in single-factor change, and multi-factor change conditions compared to state-of-the-art methods.

1 Introduction

Disentangling representations by intrinsic factors of datasets is a crucial issue in machine learning literature (Bengio et al., 2013). In Variational Autoencoder (VAE) frameworks, a prevalent method to handle the issue is to factorize latent vector dimensions to encapsulate specific factor information (Kingma & Welling, 2013; Higgins et al., 2017; Chen et al., 2018; Kim & Mnih, 2018; Jeong & Song, 2019; Shao et al., 2020; 2022). Although their effective disentanglement learning methods, (Locatello et al., 2019) raises the serious difficulty of disentanglement without sufficient inductive bias.

In VAE literature, recent works using group theory offer a possible solution to inject such inductive bias by decomposing group symmetries (Higgins et al., 2018) in the latent vector space. To implement group equivariant VAE, Winter et al. (2022a); Nasiri & Bepler (2022) achieve the translation and rotation equivariant VAE. The other branch implements the group equivariant function (Yang et al., 2022; Keller & Welling, 2021b) over the pre-defined group elements. All of the methods effectively improve disentanglement by adjusting symmetries, but they focused on learning symmetries among observations to inject inductive bias rather than factorizing group elements to align them on a single factor and a single dimension changes, as introduced in the definition provided in Higgins et al. (2018).

In current works, unsupervised learning approaches of group equivariant models are introduced. Miyato et al. (2022); Quessard et al. (2020) represent the symmetries on the latent vector space, which correspond to the symmetries on the input space, by considering the sequential observations. Also, Winter et al. (2022b) proposes the group invariant and equivariant representations with different modules to learn the different groups of dataset structure. However, these approaches, despite being unsupervised learning, require the factor information of the dataset to construct the sequential input and to set different modules for learning symmetries.

This paper introduces a novel disentanglement method for Composite Factor-Aligned Symmetry Learning (CFASL) within VAE frameworks, aimed at addressing the challenges encountered in unsupervised learning scenarios, particularly the absence of explicit knowledge about the factor structure in datasets. Our methodology follow as 1) a network architecture to learn an explicit codebook of symmetries, responsible for each single factor change, called *factor-aligned* symmetries, 2) training losses to inject inductive bias to be an explicit codebook where each factor-aligned symmetry only impacts to a single dimension value of latent vectors for disentangled representations, 3) learning composite symmetries by prediction of single factor changes itself without information of factor labels for unsupervised learning, 4) implementing group equivariant encoder and decoder functions that factor-aligned symmetry affects latent vector space, 5) an extended metric (m-FVM$_k$) to evaluate disentanglement in the multi-factor change condition. We conduct quantitative and qualitative analyses of our method onn common benchmarks of disentanglement in VAEs.

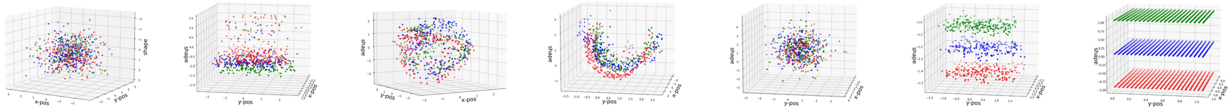

(a) β-VAE (b) β-TCVAE (c) Cont.-VAE (d) CLG-VAE (e) G-VAE (f) Ours (g) Ideal

Figure 1: Distribution of latent vectors for dimensions responsible for Shape, X-pos, Y-pos factors in the dSprites dataset. The groupified-VAE method is applied to β-TCVAE because this model shows a better evaluation score. The results show disentanglement for Shape from the combination of the other two factors by coloring three shapes (square, ellipse, heart) as red, blue, and green color, respectively. Each 3D-plot shows the whole distribution. We fix Scale and Orientation factor values, and plot randomly sampled 640 inputs (20.8% of all possible observations $(32 \times 32 \times 3 = 3,072)$). We select the dimensions responsible for the factors by selecting the largest value of the Kullback-Leibuler divergence between the prior and the posterior. Cont.-VAE is a Control-VAE.

2 Preliminaries

2.1 Group Theory

Group: A group is a set G together with binary operation $\circ$, that combines any two elements g_a and g_b in G, such that the following properties:

- closure: $g_a, g_b \in G \Rightarrow g_a \circ g_b \in G$.
- Associativity: $\forall g_a, g_b, g_c \in G,\ s.t.\ (g_a \circ g_b) \circ g_c = g_a \circ (g_b \circ g_c)$.
- Identity element: There exists an element $e \in G,\ s.t.\ \forall g \in G, e \circ g = g \circ e = g$.
- Inverse element: $\forall g \in G, \exists g^{-1} \in G\text{:}\ g \circ g^{-1} = g^{-1} \circ g = e$.

Group action: Let $(G, \circ)$ be a group and set X, binary operation $\cdot : G \times X \rightarrow X$, such that following properties:

- Identity: $e \cdot x = x$, where $e \in G, x \in X$.
- Compatibility: $\forall g_a, g_b \in G, x \in X, (g_a \circ g_b) \cdot x = g_a \cdot (g_b \cdot x)$.

Equivariant map: Let G be a group and X_1, X_2 be two sets with corresponding group action of G in each sets: $T_g^{X_1}, T_g^{X_2}$, where $g \in G$. Then a function $f : X_1 \rightarrow X_2$ is equivariant if $f(T_g^{X_1} \cdot X_1) = T_g^{X_2} \cdot f(X_1)$.

2.2 Symmetries for Inductive Bias

To implement an equivariant map for learning symmetries, previous works utilize the Variational Auto-Encoder with an additional objective function defined as $q_\phi(g^x \cdot_x x) = g^z \cdot_z q_\phi(x)$, where $q_\phi : x \rightarrow z$ is an encoder, g^x, g^z is a symmetry on input space $\mathcal{X}$ and latent vector space $\mathcal{Z}$, respectively. Yang et al. (2022); Winter et al. (2022b), represent the g^z in the latent vector space instead g^x with following equation: $q_\phi(x_2) = g^z \cdot_z q_\phi(x_1)$, where $x_2 = g^x \cdot_x x_1$ to represent the g^z correspond to g^x. Symmetry g^z is defined as a specific group such as $SO(3), SE(n)$ or $GL(n)$ (Yang et al., 2022; Winter et al., 2022b; Zhu et al., 2021).

3 Limits of Disentanglement Learning of VAE

By the definition of disentangled representation (Bengio et al., 2013; Higgins et al., 2018), the disentangled representations distribute on the flattened surface as shown in Fig. 1g because each change of the factor only affects to single dimension of latent vector. However, the previous methods (Higgins et al., 2017; Chen et al., 2018; Shao et al., 2020; Zhu et al., 2021; Yang et al., 2022) show the entangled representations on their latent vector space as shown in Fig. 1a-1c. Even though the group theory-based methods improve the disentanglement performance (Zhu et al., 2021; Yang et al., 2022), these are still struggling with the same problem as shown in Fig. 1d and 1e. In addition, symmetries are represented on the latent vector space for disentangled representations. In current works (Miyato et al., 2022; Keller & Welling, 2021b; Quessard et al., 2020), the sequential observation is considered with unsupervised learning. However, these works need the knowledge of sequential changes of images to set up inputs manually.

To enhance these two problems of disentanglement learning of group theory-based methods, addressing two questions is crucial:

1. Do the explicitly defined symmetries impact to the structuring of a disentangled space as depicted in Fig. 1g?
2. Can these symmetries be represented through unsupervised learning without any prior knowledge of factor information?

4 Methods

Our work is mainly focused on how to 1) extract the transformation between two inputs, 2) define the symmetries on the latent vector space, and 3) inject the symmetry as an inductive bias. We first introduce paired inputs in section 4.1 to extract the symmetry. In the next, we define the symmetries on the latent vector and set the learnable explicit symmetry codebook to represent the symmetry on the latent vector space in section 4.2. In the last, we introduce the objective loss to inject an inductive bias for disentangled representations in the section 4.2 and 4.3.

4.1 Input: A Pair of Two Samples

To learn the symmetries between inputs with unknown factors changes, we randomly pair the two samples as an input. During the training, samples in the mini-batch $\mathbb{X}_{|B|}$ are divided into two parts $\mathbb{X}^1_{|B|} = \{x^1_1, x^1_2, \ldots, x^1_{\frac{|B|}{2}}\}$, and $\mathbb{X}^2_{|B|} = \{x^2_1, x^2_2, \ldots, x^2_{\frac{|B|}{2}}\}$, where $|B|$ is a mini-batch size. In the next, our model pairs the samples $(x^1_1, x^2_1), (x^1_2, x^2_2), \ldots, (x^1_{\frac{|B|}{2}}, x^2_{\frac{|B|}{2}})$ and is used for learning symmetries between the elements of each pair.

4.2 Factor-Aligned Symmetry Learning with Inductive Bias

We define the *factor-aligned symmetry* that represents a corresponding factor change on the latent vector space and each independent factor only affects a single dimension value of latnet vector. For factor-aligned symmetry, we compose the symmetry codebook and inject inductive bias via Parallel loss $\mathcal{L}_{pl}$ and Perpendicular loss $\mathcal{L}_{pd}$ that matches each symmetry to a single factor changes. Then we add sparsity loss $\mathcal{L}_s$ to the losses for disentangled representations as shown in Fig. 3. It aligns a single factor change to an axis of latent vector space. Also, we implement the commutative loss $\mathcal{L}_c$ to reduce the computational costs for matrix exponential multiplication.

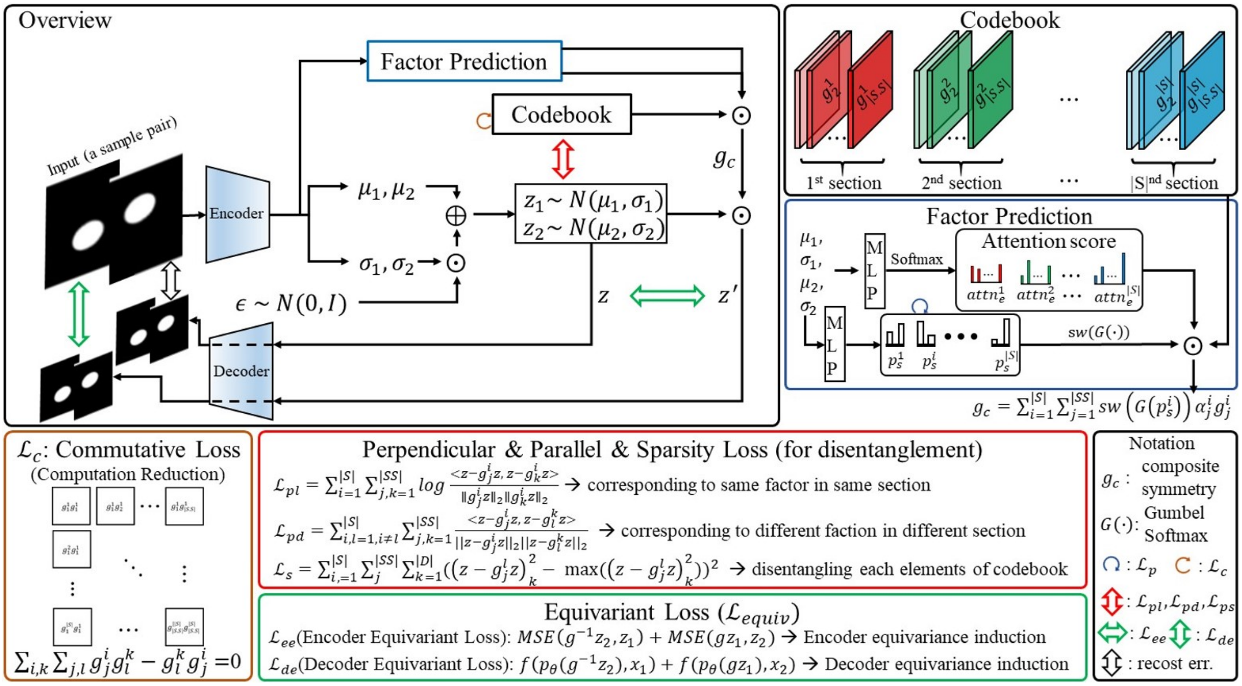

Overview
Factor Prediction
Codebook
Input (a sample pair)
Encoder
μ_1, μ_2
σ_1, σ_2
$z_1 \sim N(\mu_1, \sigma_1)$
$z_2 \sim N(\mu_2, \sigma_2)$
$\epsilon \sim N(0, I)$
g_c
z
z'
Decoder
Codebook
g^1_2 $g^1_{|S,S|}$
g^2_2 $g^2_{|S,S|}$
$g^{|S|}_2$ $g^{|S|}_{|S,S|}$
1st section
2nd section
|S|nd section
Factor Prediction
$\mu_1, \sigma_1, \mu_2, \sigma_2$
MLP
Softmax
Attention score
$attn^1_e$ $attn^2_e$ $attn^{|S|}_e$
p^1_s p^i_s $p^{|S|}_s$
$sw(G(\cdot))$
$g_c = \sum_{i=1}^{|S|}\sum_{j=1}^{|SS|} sw\left(G(p^i_s)\right)\alpha^i_j g^i_j$
$\mathcal{L}_c$: Commutative Loss
(Computation Reduction)
$g^1_1 g^1_1$ $g^1_1 g^1_2$ $g^1_1 g^1_{|S,S|}$ $g^2_1 g^1_1$ $g^{|S|}_1 g^1_1$ $g^{|S|}_{|S,S|} g^{|S|}_{|S,S|}$
$\sum_{i,k}\sum_{j,l} g^i_j g^k_l - g^k_l g^i_j = 0$
Perpendicular & Parallel & Sparsity Loss (for disentanglement)
$\mathcal{L}_{pl} = \sum_{i=1}^{|S|}\sum_{j,k=1}^{|SS|} log \frac{<z-g^i_j z, z-g^i_k z>}{\|g^i_j z\|_2 \|g^i_k z\|_2}$ → corresponding to same factor in same section
$\mathcal{L}_{pd} = \sum_{i,l=1,i\neq l}^{|S|}\sum_{j,k=1}^{|SS|} \frac{<z-g^i_j z, z-g^k_l z>}{||z-g^i_j z||_2 ||z-g^k_l z||_2}$ → corresponding to different faction in different section
$\mathcal{L}_s = \sum_{i,=1}^{|S|}\sum_{j}^{|SS|}\sum_{k=1}^{|D|}((z-g^l_j z)^2_k - \max((z-g^l_j z)^2_k))^2$ → disentangling each elements of codebook
Equivariant Loss ($\mathcal{L}_{equiv}$)
$\mathcal{L}_{ee}$(Encoder Equivariant Loss): $MSE(g^{-1}z_2, z_1) + MSE(gz_1, z_2)$ → Encoder equivariance induction
$\mathcal{L}_{de}$(Decoder Equivariant Loss): $f(p_\theta(g^{-1}z_2), x_1) + f(p_\theta(gz_1), x_2)$ → Decoder equivariance induction
Notation
g_c : composite symmetry
$G(\cdot)$: Gumbel Softmax
: $\mathcal{L}_p$
: $\mathcal{L}_c$
: $\mathcal{L}_{pl}, \mathcal{L}_{pd}, \mathcal{L}_{ps}$
: $\mathcal{L}_{ee}$
: $\mathcal{L}_{de}$
: recost err.

Figure 2: The overall architecture of our proposed method. The loss function is divided into four parts: 1) commutative loss ($\mathcal{L}_c$), 2) perpendicular, parallel, and sparsity loss ($\mathcal{L}_{pd}$, $\mathcal{L}_{pl}$, and $\mathcal{L}_s$) in Equation 1-3, 3) factor prediction loss ($\mathcal{L}_p$) in Equation 4), and 4) equivariant loss ($\mathcal{L}_{ee}$, and $\mathcal{L}_{de}$) in Equation 8). MLP is a multi-layer perceptron, and, tr is a threshold. Attention score $attn_e$, $sw(\cdot)$, and p^i_s are introduced in section 4.3.

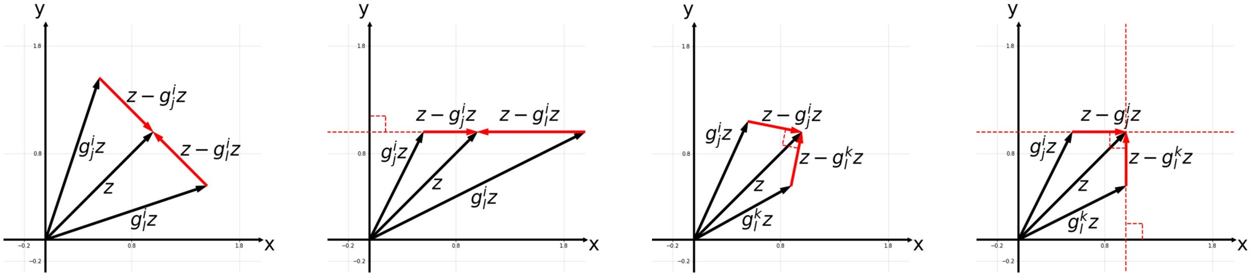
Explicit and Learnable Symmetry Representation for Inductive Bias Injection To allow the direct injection of inductive bias into symmetries, we implement an explicit and trainable codebook for symmetry representation. we consider the symmetry group on the latent vector space as a subgroup of the general lie group $GL(n)$ under a matrix multiplication. The codebook $\mathcal{G} = \{\mathcal{G}^1, \mathcal{G}^2, \dots, \mathcal{G}^k\}$ is composed of sections $\mathcal{G}^i$, which are affect to a different single factor, where $k \in \{1, 2, \dots |S|\}$, and $|S|$ is the number of sections. The section $\mathcal{G}^i$ is composed of Lie algebra $\{\mathfrak{g}^i_1, \mathfrak{g}^i_2, \dots, \mathfrak{g}^i_l\}$, where $\mathfrak{g}^i_j \in \mathbb{R}^{|D|\times|D|}$, $l \in \{1, 2, \dots, |SS|\}$, $|SS|$ is the number of elements in each section, and $|D|$ is a dimension size of latent $\boldsymbol{z}$. We assume that each Lie algebra consists of linearly independent bases $\mathfrak{B} = \{\mathfrak{B}_i | \mathfrak{B}_i \in \mathbb{R}^{n\times n}, \sum_i \alpha_i \mathfrak{B}_i \neq 0, \alpha_i \neq 0\}$: $\mathfrak{g}^i_j = \sum_b \alpha^{i,j}_b \mathfrak{B}_b$, where $b \in \{1, 2, \dots, kl\}$. Then the dimension of the element of the codebook is equal to $|\mathfrak{B}|$ and the dimension of the Lie group composited by the codebook element is also $|\mathfrak{B}|$. To utilize previously studied effective expression of symmetry for disentanglement, we set the symmetry to be continuous (Higgins et al., 2022) and invertible via matrix exponential form (Xiao & Liu, 2020) as $g^i_j = \mathbf{e}^{\mathfrak{g}^i_j} = \sum_{k=0}^{\infty} \frac{1}{k!}(\mathfrak{g}^i_j)^k$ to construct the Lie group (Hall, 2015).

Inductive Bias: Group Elements of the Same Section Impacts on the Same Factor Changes We add a bias that latent vector changes by symmetries of the same section to be parallel to contain the same factor information ($\boldsymbol{z} - g^i_j\boldsymbol{z} \parallel \boldsymbol{z} - g^i_l\boldsymbol{z}$ for ith section) as shown in Fig. 3a. We define a loss function to make them parallel as:

$$\mathcal{L}_{pl} = \sum_{i=1}^{|S|}\sum_{j,k=1}^{|SS|} \log \frac{<\boldsymbol{z} - g^i_j\boldsymbol{z}, \boldsymbol{z} - g^i_k\boldsymbol{z}>}{||\boldsymbol{z} - g^i_j\boldsymbol{z}||_2 \cdot ||\boldsymbol{z} - g^i_k\boldsymbol{z}||_2}, \tag{1}$$

where $g^i_j = \mathbf{e}^{\mathfrak{g}^i_j}$, $< \cdot, \cdot >$ is a dot product, and $|| \cdot ||_2$ is a L2 norm.

Inductive Bias: Group Elements of the Different Section Impacts on the Different Factor Changes Similarly to the parallel loss, we inject another bias that changes by symmetries of the different section to be orthogonal for different factors change impacts ($\boldsymbol{z} - g^i_j\boldsymbol{z} \perp \boldsymbol{z} - g^k_l\boldsymbol{z}$ for different ith and kth



(a) parallel loss (b) (a) + sparsity loss (c) perpendicular loss (d) (c) + sparsity loss

Figure 3: Roles of parallel, perpendicular, and sparsity loss on symmetries in the codebook for adjusting representation change. Parallel loss is for symmetries of the same section, and perpendicular loss is for different sections. Each axis (x and y) only affects to a single factor.

sections) as shown in Fig. 3c. The loss for inducing the orthogonality is

$$\mathcal{L}_{pd} = \sum_{i,k=1, i\neq k}^{|S|} \sum_{j,l=1}^{|SS|} \frac{< \boldsymbol{z} - g_j^i \boldsymbol{z}, \boldsymbol{z} - g_l^k \boldsymbol{z} >}{||\boldsymbol{z} - g_j^i \boldsymbol{z}||_2 \cdot ||\boldsymbol{z} - g_l^k \boldsymbol{z}||_2}. \tag{2}$$

Expensive computational cost for Eq. 2 ($O(|S|^2 \cdot |SS|^2)$), we randomly select a (j, l) pair of symmetries of different sections. This random selection still holds the orthogonality, because if all elements in the same section satisfy Equation 1 and a pair of elements from a different section $(\mathcal{G}^i, \mathcal{G}^j)$ satisfies Equation 2, then any pair of the element $(\mathfrak{g}^i \in \mathcal{G}^i, \mathfrak{g}^j \in \mathcal{G}^j)$ satisfies the Equation 2. More details are in the Appendix B.

Inductive Bias: Align Each Factor Changes to the Axis of Latent Space for Disentangled Representations Factorizing latent dimensions to represent the change of independent factors is an attribute of disentanglement defined in Bengio et al. (2013) and derived by ELBO term in VAE training frameworks (Chen et al., 2018; Kim & Mnih, 2018). However, it is not worthy that neither the parallel loss nor the perpendicular loss exhibits specific constraints aimed at inducing this characteristic as shown in Fig. 3a, 3c. To guide symmetries to hold the attribute, we enforce the change $\Delta_j^i \boldsymbol{z} = \boldsymbol{z} - g_j^i \boldsymbol{z}$ to be a parallel shift to a unit vector as Fig. 3b, 3d via sparsity loss defined as

$$\mathcal{L}_s = \sum_{i=1}^{|S|} \sum_{j}^{|SS|} \Big[\big[\sum_{k=1}^{|D|} (\Delta_j^i \boldsymbol{z}_k)^2 \big]^2 - \max_k ((\Delta_j^i \boldsymbol{z}_k)^2)^2 \Big], \tag{3}$$

where $\Delta_j^i \boldsymbol{z}_k$ is a k^{th} dimension value.

4.3 Composition of Factor-Aligned Symmetries via Two-Step Attention for Unsupervised Learning

First Step: Select Factor-Aligned Symmetry In the first step, the model generates the factor-aligned symmetries of each section through the attention score as shown in Fig. 2: $g_c^i = \boldsymbol{e}^{\mathfrak{g}_c^i}$, where $\mathfrak{g}_c^i = \sum_{j=1}^{|SS|} attn_j^i \mathfrak{g}_j^i$, $attn_j^i = softmax([M;\Sigma]\boldsymbol{W}_c^i + \boldsymbol{b}_c^i)$ $[M;\Sigma] = [\boldsymbol{\mu}_1; \boldsymbol{\sigma}_1; \boldsymbol{\mu}_2; \boldsymbol{\sigma}_2]$, $\boldsymbol{W}_c^i \in \mathbb{R}^{4|D| \times |SS|}$ and $\boldsymbol{b}_c^i \in \mathbb{R}^{|SS|}$ are learnable parameters, and $i \in \{1, 2, \dots |S|\}$.

Second Step: Section Selection In the second step of our proposed model, we enforce the prediction of factors deemed to have undergone changes. We assume that if some factor value of two inputs is equal, then the variance of the corresponding latent vector dimension value is smaller compared to others. Based on this assumption, we define the target (T) for factor prediction: if $\boldsymbol{z}_{1,i} - \boldsymbol{z}_{2,i} >$ threshold, then we set T_i as 1 and 0 otherwise, where T_i is a i^{th} dimension value of $T \in \mathbb{R}^{|D|}$, $\boldsymbol{z}_{j,i}$ is an i^{th} dimension value of $\boldsymbol{z}_j$, and we set the threshold as a hyper-parameter. For section prediction, we utilize the cross-entropy loss:

$$\mathcal{L}_p = \sum_{i=1}^{|S|} \sum_{c \in C} \nVdash [T_i = c] \cdot \log softmax(p_s^i), \tag{4}$$

where $p_s^i = [M; \Sigma]\boldsymbol{W}_s^i + \boldsymbol{b}_s^i$, $\boldsymbol{W}_s^i \in \mathbb{R}^{4|D|\times 2}$ and $\boldsymbol{b}_s^i \in \mathbb{R}^2$ are learnable parameters, and $c \in \{0, 1\}$.

To infer the activated section of the symmetries codebook, we utilize the Gumbel softmax function to handle binary on-and-off scenarios, akin to a switch operation:

$$sw(G(p_s^i)) = \begin{cases} G(p_{s,2}^i) & \text{if } p_{s,2}^i \geq 0.5 \\ 1 - G(p_{s,1}^i) & \text{if } p_{s,2}^i < 0.5 \end{cases}, \tag{5}$$

where $p_{s,j}^i$ is a j^{th} dimension value of p_s^i, and $G(\cdot)$ is the Gumbel softmax with temperature as 1e-4.

Integration for Composite Symmetry For the composite symmetry g_c, we compute the product of weighted sums of switch function $sw(p_s)$ and prediction distribution $attn$ as: $g_c = \prod_{i=1}^{|S|} \hat{g}_c^i$, where $\hat{g}_c^i = \mathbf{e}^{sw(G(p_s^i))\cdot \mathfrak{g}_c^i}$.

Commutativity Loss for Computational Efficiency In the computation of the composite symmetry g_c, the production $\prod_{i=1}^{|S|} \hat{g}_c^i$ is a computationally expensive Taylor series repeated for all (i, j) pairs. To reduce the cost by repetition, we enforce all pairs of basis $\mathfrak{g}_i^j$ to be commutative to convert the production to $\mathbf{e}^{\sum_i \mathfrak{g}_c^i}$ (By the matrix exponential property: $\mathbf{e}^{\boldsymbol{A}}\mathbf{e}^{\boldsymbol{B}} = \mathbf{e}^{\boldsymbol{A}+\boldsymbol{B}}$ as $\boldsymbol{AB} = \boldsymbol{BA}$, where $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$). The loss for the commutativity is $\mathcal{L}_c = \sum_{i,k=1}^{|S|} \sum_{j,l=1}^{|SS|} \mathfrak{g}_j^i \mathfrak{g}_l^k - \mathfrak{g}_l^k \mathfrak{g}_j^i \rightarrow \mathbf{0}$.

4.4 Equivariance Induction of Composite Symmetries

How to Induce Equivariance? Motivated by the implementations of equivariant mapping in prior studies (Yang et al., 2022; Miyato et al., 2022) for disentanglement learning, we implement an equivariant encoder and decoder that satisfies $q_\phi(\psi_i * x) = g_i \circ q_\phi(x)$ and $p_\theta(g_i \circ z) = \psi_i * p_\theta(z)$ respectively, where q_ϕ is an encoder, and p_θ is the decoder. In the notation, ψ_i and g_i are group elements of the group $(\Psi, *)$ and $(\mathcal{G}, \circ)$ respectively, and both groups are isomorphic. Each group acts on the input and latent vector space with group action $*$, and $\circ$, respectively. We specify the form of symmetry g_i and $\circ$ as an invertible matrix, and group action as matrix multiplication on the latent vector space. Then, the encoder equivariant function can be rewritten by multiplying the inversion of g_i on both sides and z can be replaced with the $q_\phi(x)$ in the decoder equivariant function as

$$\begin{aligned} &q_\phi(x) = g_i^{-1} \circ q_\phi(\psi_i * x) \\ &\iff q_\phi(x) - g_i^{-1} \circ q_\phi(\psi_i * x) \rightarrow 0 \qquad \text{(for encoder)}. \end{aligned} \tag{6}$$

$$\begin{aligned} &p_\theta(g_i \circ q_\phi(x)) = \psi_i * p_\theta(q_\phi(x)) \\ &\iff p_\theta(g_i \circ q_\phi(x)) - \psi_i * p_\theta(q_\phi(x)) \rightarrow 0 \text{ (for decoder)}, \end{aligned} \tag{7}$$

where $x_j = \psi_{i\rightarrow j} * x_i$. For the equivariant encoder and decoder, we differently propose the single forward process by the encoder and decoder objective functions compared to previous work (Yang et al., 2022).

Equivariance Loss for Encoder and Decoder In order for the equivariant function between the input and latent vector space, the mapping function $q_\phi(\cdot)$ must satisfy Equation 6. Therefore, we directly induce an equivariant encoder between input and latent space with MSE loss ($\mathcal{L}_{ee}$). Additionally, we induce the equivariant decoder ($\mathcal{L}_{de}$) with MSE loss following Equation 7:

$$\begin{aligned} \mathcal{L}_{equiv} = \mathcal{L}_{ee} + \epsilon\mathcal{L}_{de} &= \text{MSE}(q_\phi(x_i^1), g_i^{-1} \circ q_\phi(x_i^2)) \\ &+ \epsilon\text{MSE}(p_\theta(g_i \circ (q_\phi(x_i^1)), \psi_i * p_\theta(q_\phi(x_i^1)), \end{aligned} \tag{8}$$

where $x_i^2 = \psi_i * x_i^1$. During the training, we replace the $p_\theta(q_\phi(x_i^1))$ as a x_i^1 because the ELBO term includes the reconstruction error between $p_\theta(q_\phi(x_i^1))$ and x_i^1 to be close to zero.

Objective and Base model Our method can be plugged into existing VAE frameworks, where the objective function is integrated additively as follows:

$$\mathcal{L}(\phi, \theta; \boldsymbol{x}) = \mathcal{L}_{VAE} + \mathcal{L}_{codebook} + \mathcal{L}_{equiv}, \tag{9}$$

where $\mathcal{L}_{VAE}$ is the loss function of a VAE framework (Appendix A). The other loss $\mathcal{L}_{codebook} = \mathcal{L}_{pl} + \mathcal{L}_{pd} + \mathcal{L}_s + \mathcal{L}_c + \mathcal{L}_p$ and $\mathcal{L}_{equiv} = \mathcal{L}_{ee} + \epsilon\mathcal{L}_{de}$ where ϵ is a hyper-parameter, which are introduced in the following subsections.

4.5 Extended Evaluation Metric: m-FVM Metric for Disentanglement in Multi-Factor Change

We define the *multi-factor change* condition as simultaneously altering more than two factors in the transformation between two samples or representations. To the best of our knowledge, there is no evaluation metric for disentanglement in multi-factor change, so we propose the extended version of the Factor-VAE metric (FVM) score called as multi-FVM score (m-FVM$_k$), where $k \in \{2, 3, \ldots, |F| - 1\}$, and $|F|$ is a number of factors. Similar to FVM, 1) we randomly choose the k fixed factors $(F_i, F_j, \ldots)$. 2) We sample each factor's value $(f_i, f_j, \ldots)$ and fix the corresponding factor dimension value in the mini-batch, where $f_i \in \{1, 2, \ldots |F_i|\}, f_j \in \{1, 2, \ldots |F_j|\}, \ldots$, $|F_i|$ and $|F_j|$ is a maximum value of each factor label. 3) Subsequently, we estimate the standard deviation (std.) of each dimension to find the number of k lowest std. dimension $(z_{l1}, z_{l2}, \ldots)$ in one epoch. 4) We then count each pair of selected dimensions by std. values (the number of $(z_{l1}, z_{l2}, \ldots)$, which are corresponded to fixed factors). 5) In the last, we add the maximum value of the number of $(z_{l1}, z_{l2}, \ldots)$ on all fixed factor cases, and divide with epoch.

5 Related Work

Disentanglement Learning Diverse works for unsupervised disentanglement learning have elaborated in the machine learning field. The VAE based approaches have factorized latent vector dimensions with weighted hyper-parameters or controllable weighted values to penalize Kullback-Leibler divergence (KL divergence) (Higgins et al., 2017; Shao et al., 2020; 2022). Extended works penalize total correlation for factorizing latent vector dimensions with divided KL divergence (Chen et al., 2018) and discriminator (Kim & Mnih, 2018). Differently, we induce disentanglement learning with group equivariant VAE for inductive bias.

Group Theory-Based Approaches for Disentangled Representation In recent periods, various unsupervised disentanglement learning research proposes different approaches with another definition of disentanglement, which is based on the group theory (Higgins et al., 2018). To learn the equivariant function, Topographic VAE (Keller & Welling, 2021a) proposes the sequentially permuted activations on the latent vector space called shifting temporal coherence, and Groupified VAE (Yang et al., 2022) method proposes that inputs pass the encoder and decoder two times to implement permutation group equivariant VAE models. Also, Commutative Lie Group VAE (CLG-VAE) (Zhu et al., 2021; Mercatali et al., 2022) maps latent vectors into Lie algebra with one-parameter subgroup decomposition for inductive bias to learn the group structure from abstract canonical point to inputs. Differently, we propose the trainable symmetries that are extracted between two samples directly on the latent space while maintaining the equivariance function between input and latent vector space.

Symmetry Learning with Equivariant Model Lie group equivariant CNN (Dehmamy et al., 2021; Finzi et al., 2020) construct the In the other literature, several works extract symmetries, which consist of matrices, between two inputs or objects. Miyato et al. (2022) extracts the symmetries between sequential or sequentially augmented inputs by penalizing the transformation of difference of the same time interval. Other work extracts the symmetries by comparing two inputs, in which the differentiated factor is a rotation or translation, and implements symmetries with block diagonal matrices (Bouchacourt et al., 2021). Furthermore, Marchetti et al. (2023) decomposes the class and pose factor simultaneously by invariant and equivariant loss function with weakly supervised learning. The unsupervised learning work (Winter et al., 2022a) achieves class invariant and group equivariant function in less constraint conditions. Differently, we generally extend the a class invariant and group equivariant model in the more complex disparity condition without any knowledge of the factors of datasets.

Table 1: Disentanglement scores for single factor change (left 5 metrics) and multi-factor change (m-FVMs) with 10 random seeds.

3D Car	FVM	beta VAE	MIG	SAP	DCI	m-FVM$_2$	m-FVM$_3$	m-FVM$_4$
β-VAE	91.83(±4.39)	100.00(±0.00)	11.44(±1.07)	0.63(±0.24)	27.65(±2.50)	61.28(±9.40)	-	-
β-TCVAE	92.32(±3.38)	100.00(±0.00)	17.19(±3.06)	1.13(±0.37)	33.63(±3.27)	59.25(±5.63)	-	-
Factor-VAE	93.22(±2.86)	100.00(±0.00)	10.84(±0.93)	1.35(±0.48)	24.31(±2.30)	50.43(±10.65)	-	-
Control-VAE	93.86(±5.22)	100.00(±0.00)	9.73(±2.24)	1.14(±0.54)	25.66(±4.61)	46.42(±10.34)	-	-
CLG-VAE	91.61(±2.84)	100.00(±0.00)	11.62(±1.65)	1.35(±0.26)	29.55(±1.93)	47.75(±5.83)	-	-
CFASL	**95.70**(±1.90)	**100.00**(±0.00)	**18.58**(±1.24)	**1.43**(±0.18)	**34.81**(±3.85)	**62.43**(±8.08)	-	-

smallNORB	FVM	beta VAE	MIG	SAP	DCI	m-FVM$_2$	m-FVM$_3$	m-FVM$_4$
β-VAE	60.71(±2.47)	59.40(±7.72)	21.60(±0.59)	11.02(±0.18)	25.43(±0.48)	24.41(±3.34)	15.13(±2.76)	-
β-TCVAE	59.30(±2.52)	60.40(±5.48)	21.64(±0.51)	11.11(±0.27)	25.74(±0.29)	25.71(±3.51)	15.66(±3.74)	-
Factor-VAE	61.93(±1.90)	56.40(±1.67)	**22.97**(±0.49)	11.21(±0.49)	24.84(±0.72)	26.43(±3.47)	17.25(±3.50)	-
Control-VAE	60.63(±2.67)	61.40(±4.33)	21.55(±0.53)	11.18(±0.48)	**25.97**(±0.43)	24.11(±3.41)	16.12(±2.53)	-
CLG-VAE	62.27(±1.71)	62.60(±5.17)	21.39(±0.67)	10.71(±0.33)	22.95(±0.62)	27.71(±3.45)	17.16(±3.12)	-
CFASL	**62.73**(±3.96)	**63.20**(±4.13)	22.23(±0.48)	**11.42**(±0.48)	24.58(±0.51)	**27.96**(±3.00)	**17.37**(±2.33)	-

dSprites	FVM	beta VAE	MIG	SAP	DCI	m-FVM$_2$	m-FVM$_3$	m-FVM$_4$
β-VAE	73.54(±6.47)	83.20(±7.07)	13.19(±4.48)	5.69(±1.98)	21.49(±6.30)	53.80(±10.29)	50.13(±11.98)	48.02(±8.98)
β-TCVAE	79.19(±5.87)	89.20(±4.73)	23.95(±10.13)	7.20(±0.66)	35.33(±9.07)	61.75(±6.71)	57.82(±5.39)	63.81(±9.45)
Factor-VAE	78.10(±4.45)	84.40(±5.55)	25.74(±10.58)	6.37(±1.82)	32.30(±9.47)	58.39(±5.18)	51.63(±2.88)	53.71(±4.22)
Control-VAE	69.64(±7.67)	82.80(±7.79)	5.93(±2.78)	3.89(±1.89)	12.42(±4.95)	38.99(±9.31)	29.00(±10.75)	19.33(±5.98)
CLG-VAE	**82.33**(±5.59)	86.80(±3.43)	23.96(±6.08)	7.07(±0.86)	31.23(±5.32)	63.21(±8.13)	48.68(±9.59)	51.00(±8.13)
CFASL	82.30(±5.64)	**90.20**(±5.53)	**33.62**(±8.18)	**7.28**(±0.63)	**46.52**(±6.18)	**68.32**(±0.13)	**66.25**(±7.36)	**71.35**(±12.08)

3D Shapes	FVM	beta VAE	MIG	SAP	DCI	m-FVM$_2$	m-FVM$_3$	m-FVM$_4$
β-VAE	84.33(±10.65)	91.20(±4.92)	45.80(±21.20)	8.66(±3.80)	66.05(±7.44)	70.26(±6.27)	61.52(±8.62)	60.17(±8.48)
β-TCVAE	86.03(±3.49)	87.80(±3.49)	60.02(±10.05)	5.88(±0.79)	70.38(±4.63)	70.20(±4.08)	63.79(±5.66)	**63.61**(±5.90)
Factor-VAE	79.54(±10.72)	95.33(±5.01)	52.68(±22.87)	6.20(±2.15)	61.37(±12.46)	66.93(±17.49)	63.55(±18.02)	57.00(±21.36)
Control-VAE	81.03(±11.95)	95.00(±5.60)	19.61(±12.53)	4.76(±2.79)	55.93(±13.11)	62.22(±11.35)	55.83(±13.61)	51.66(±12.08)
CLG-VAE	83.16(±8.09)	89.20(±4.92)	49.72(±16.75)	6.36(±1.68)	63.62(±3.80)	65.13(±5.26)	58.94(±6.59)	60.51(±7.62)
CFASL	**89.70**(±9.65)	**96.20**(±4.85)	**62.12**(±13.38)	**9.28**(±1.92)	**75.49**(±8.29)	**74.26**(±2.82)	**67.68**(±2.67)	63.48(±4.12)

Table 2: Disentanglement performance rank. Each dataset rank is an average of evaluation metrics, and Avg. is an average of all datasets.

	3D Car	smallNORB	dSprites	3D Shapes	Avg.
β-VAE	3.33	4.86	4.88	3.38	4.11
β-TCVAE	2.50	4.29	2.50	2.88	3.04
Factor-VAE	3.17	2.71	3.38	4.00	3.31
Control-VAE	3.67	3.86	6.00	5.50	4.76
CLG-VAE	3.00	3.86	3.13	4.13	3.53
CFASL	**1.00**	**1.43**	**1.13**	**1.13**	**1.17**

6 Experiments

Device We set the below settings for all experiments in a single Galaxy 2080Ti GPU for 3D Cars and smallNORB, and a single Galaxy 3090 for dSprites 3D Shapes and CelebA. More details are in README.md file.

Datasets 1) The dSprites dataset consists of 737,280 binary 64×64 images with five independent ground truth factors(number of values), *i.e.* x-position(32), y-position(32), orientation(40), shape(3), and scale(6), Matthey et al. (2017). Any composite transformation of x- and y-position, orientation (2D rotation), scale, and shape is commutative. 2) The 3D Cars dataset consists of 17,568 RGB $64 \times 64 \times 3$ images with three independent ground truth factors: elevations(4), azimuth directions(24), and car models(183) Reed et al. (2015). Any composite transformation of elevations(x-axis 3D rotation), azimuth directions (y-axis 3D rotation), and models are commutative. 3) The smallNORB LeCun et al. (2004) dataset consists of total 96×96 24,300 grayscale images with four factors, which are category(10), elevation(9), azimuth(18), light(6) and we resize the input as 64×64. Any composite transformation of elevations(x-axis 3D rotation), azimuth (y-axis 3D rotation), light, and category is commutative. 4) The 3D Shapes dataset consists of 480,000 RGB $64 \times 64 \times 3$ images with six independent ground truth factors: orientation(15), shape(4), floor color(10), scale(8), object color(10), and wall color(10) Burgess & Kim (2018). 5) The CelebA dataset Liu et al. (2015) consists of 202,599 images, and we crop the center 128×128 area and then, resize to 64×64 images.

Evaluation Settings We set *prune_dims.threshold* as 0.06, 100 samples to evaluate global empirical variance in each dimension, and run it a total of 800 times to estimate the FVM score introduced in Kim & Mnih (2018). For the other metrics, we follow default values introduced in Michlo (2021), training and evaluation 10,000 and 5,000 times with 64 mini-batches, respectively Cao et al. (2022).

Model Hyper-parameter Tuning We implement β-VAE Higgins et al. (2017), β-TCVAE Chen et al. (2018), control-VAE Shao et al. (2020), Commutative Lie Group VAE (CLG-VAE) Zhu et al. (2021), and Groupified-VAE (G-VAE) Yang et al. (2022) for baseline. For common settings to baselines, we set batch size 64, learning rate 1e-4, and random seed from $\{1, 2, \ldots, 10\}$ without weight decay. We train for 3×10^5 iterations on dSprites smallNORB and 3D Cars, 6×10^5 iterations on 3D Shapes, and 10^6 iterations on CelebA. We set hyper-parameter $\beta \in \{1.0, 2.0, 4.0, 6.0\}$ for β-VAE and β-TCVAE, fix the α, γ for β-TCVAE as 1 Chen et al. (2018). We follow the ControlVAE settings Shao et al. (2020), the desired value $C \in \{10.0, 12.0, 14.0, 16.0\}$, and fix the $K_p = 0.01$, $K_i = 0.001$. For CLG-VAE, we also follow the settings Zhu et al. (2021) as $\lambda_{hessian} = 40.0$, $\lambda_{decomp} = 20.0$, $p = 0.2$, and balancing parameter of $loss_{\text{rec group}} \in \{0.1, 0.2, 0.5, 0.7\}$. For G-VAE, we follow the official settings Yang et al. (2022) with β-TCVAE ($\beta \in \{10, 20, 30\}$), because applying this method to the β-TCVAE model usually shows higher performance than other models Yang et al. (2022). Then we select the best case of models. We run the proposed model on the β-VAE and β-TCVAE because these methods have no inductive bias to symmetries. We set the same hyper-parameters of baselines with $\epsilon \in \{0.1, 0.01\}$, threshold $\in \{0.2, 0.5\}$, $|S| = |SS| = |D|$, where $|D|$ is a latent vector dimension. More details for experimental settings.

6.1 Quantitative Analysis Results and Discussion

Disentanglement Performance in Single and Multi-Factor Change We evaluate four common disentanglement metrics: FVM (Kim & Mnih, 2018), MIG (Chen et al., 2018), SAP (Kumar et al., 2018), and DCI (Eastwood & Williams, 2018). As shown in Table 1, our method gradually improves the disentanglement learning in dSprites, 3D Cars, 3D Shapes, and smallNORB datasets in most metrics. To show the quantitative score of the disentanglement in multi-factor change, we evaluate the m-FVM$_k$, where max(k) is 2, 3, and 4 in 3D Cars, smallNORB, and dSprites datasets respectively. These results also show that our method positively affects single and multi-factor change conditions. As shown in Table 2, the proposed method shows a statistically significant improvement, as indicated by the higher average rank of dataset metrics compared to other approaches.

Table 3: Ablation study for loss functions on 3D-Cars and β-VAE with 10 random seeds.

	$\mathcal{L}_p$	$\mathcal{L}_c$	$\mathcal{L}_{e.}$	$\mathcal{L}_{pl}$	$\mathcal{L}_{pd}$	$\mathcal{L}_s$	FVM	MIG	SAP	DCI	m-FVM$_2$
β-VAE	✗	✗	✗	✗	✗	✗	88.19(±4.60)	6.82(±2.93)	0.63(±0.33)	20.45(±3.93)	42.36(±7.16)
	✗	✓	✓	✓	✓	✓	88.57(±6.68)	7.18(±2.52)	**1.85**(±1.04)	18.39(±4.80)	48.23(±5.51)
	✓	✓	✗	✓	✓	✓	88.56(±7.78)	7.27(±4.16)	1.31(±0.70)	19.58(±4.45)	42.63(±4.21)
	✓	✓	✓	✗	✓	✓	86.95(±5.96)	7.11(±3.49)	1.09(±0.40)	18.35(±3.32)	41.90(±7.80)
	✓	✓	✓	✓	✗	✓	85.42(±7.89)	7.30(±3.73)	1.15(±0.70)	21.69(±4.70)	41.90(±6.07)
	✓	✓	✓	✗	✗	✗	89.34(±5.18)	9.44(±2.91)	1.26(±0.40)	**23.14**(±5.51)	51.37(±9.29)
	✓	✓	✓	✓	✓	✗	90.71(±5.75)	9.29(±3.74)	1.07(±0.65)	22.74(±5.06)	45.84(±7.71)
	✓	✓	✓	✓	✓	✓	**91.91**(±3.45)	**9.51**(±2.74)	1.42(±0.52)	20.72(±3.65)	**55.47**(±10.09)

Ablation Study Table 3 shows the ablation study to evaluate the impact of each component of our method for disentanglement learning. In the case of w/o $\mathcal{L}_{pl}$, the extraction of the composite symmetry g_c becomes challenging due to the lack of unified roles among individual sections. Also, the coverage of code w/o $\mathcal{L}_{pd}$ is limited due to the absence of assurance that each section aligns with distinct factors. In the case of w/o $\mathcal{L}_s$, each section assigns a different role and the elements of each section align on the same factor, and w/o $\mathcal{L}_s$ case is better than w/o $\mathcal{L}_{pl}$ and w/o $\mathcal{L}_{pd}$. These results imply that the no-differentiated role of each section struggles with constructing adequate composite symmetry g_c. Also, it shows that dividing the symmetry information in each section ($\mathcal{L}_{pl}$, $\mathcal{L}_{pd}$) is more important than $\mathcal{L}_s$ for disentangled representation. To compare factor-aligned losses (w/o $\mathcal{L}_{pl}$, w/o $\mathcal{L}_{pd}$, w/o $\mathcal{L}_s$, and w/o $\mathcal{L}_{pl} + \mathcal{L}_{pd} + \mathcal{L}_s$), the best of among four cases is the w/o $\mathcal{L}_{pl} + \mathcal{L}_{pd} + \mathcal{L}_s$ and it implies that these losses are interrelated. Also, constructing the symmetries without the equivariant model is meaningless because the model does not satisfy Equation 6-8. The w/o $\mathcal{L}_{equiv}$ naturally shows the lowest results compared to other cases except w/o $\mathcal{L}_{pd}$ and $\mathcal{L}_{pl}$. Moreover, the w/o $\mathcal{L}_p$ case shows the impact of the section selection method (section 4.3) for unsupervised learning. Above all, each group exhibits a positive influence on disentanglement when compared to the base model (β-VAE). When combining all loss functions, our method consistently outperforms the others across the majority of evaluation metrics.

Impact of Hyper-Parameter tuning We operate a grid search of the hyper-parameter ϵ. As shown in Figure 4a, the Kullback-Leibler divergence convergences to the highest value, when ϵ is large ($\epsilon = 1.0$) and it shows less stable results. It implies that the CFASL with larger ϵ struggles with disentanglement

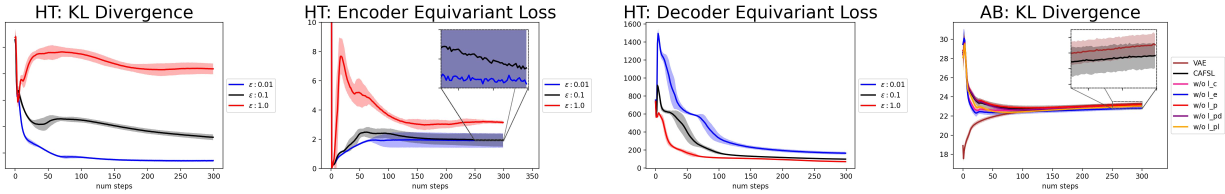
Table 4: Additional experiments

(a) Hyper-parameter tuning with 6 random seeds.

ϵ	FVM	beta VAE	MIG	SAP	DCI
0.01	76.98(±8.63)	87.33(±7.87)	29.68(±11.38)	6.96(±1.16)	41.28(±11.93)
0.1	**82.21**(±1.34)	**90.33**(±5.85)	**34.79**(±3.26)	**7.45**(±0.61)	**48.07**(±5.62)
1.0	76.77(±7.05)	78.33(±13.88)	22.42(±11.14)	6.02(±0.48)	38.87(±7.83)

(b) Codebook size impact

3D Cars	$\lvert\mathcal{G}\rvert$=100	$\lvert\mathcal{G}\rvert$=10
FVM	**95.70**(±1.90)	48.63(±24.55)
MIG	**18.58**(±1.24)	2.99(±6.04)
SAP	**1.43**(±0.18)	0.29(±0.34)
DCI	**34.81**(±3.85)	6.12(±10.44)
FVM_2	**62.43**(±8.08)	37.94(±10.01)



(a) kld (hyper) (b) encoder (hyper) (c) decoder (hyper) (d) kld (w/o)

Figure 4: Loss curves: 1) HT: hyper-parameter tuning ($\epsilon \in \{0.01, 0.1, 1.0\}$) with β-TCVAE based CFASL. 2) AB: ablation study with β-VAE based CFASL.

learning, and is shown in Tabel 4a. Also, the $\mathcal{L}_{ee}$ in Figure 4b is larger than other cases, which implies that the model struggles with extracting adequate composite symmetry because its encoder is far from the equivariant model and it is also shown in Table 4a. Even though $\epsilon = 0.01$ case shows the lowest value in the most loss, $\mathcal{L}_{de}$ in Figure 4c is higher than others and it also implies the model struggles with learning symmetries, as shown in Table 4a because the model does not close to the equivariant model compare to $\epsilon = 0.1$ case.

Posterior of CFASL The symmetry codebook and composite symmetry are linear transformations of latent vectors. intuitively, they enforce the posterior to be far from prior as $q_\phi(\boldsymbol{z}|\boldsymbol{x}) \sim \mathcal{N}(g_c\mu, g_c\Sigma g_c^\intercal)$, where μ and Σ are close to zero vectors and identity matrix respectively. However, as shown in Fig. 4d, Kullbeck Leibler divergence is lower than VAE. It represents the ability of CFASL to preserve Gaussian normal distribution, which is similar to VAE.

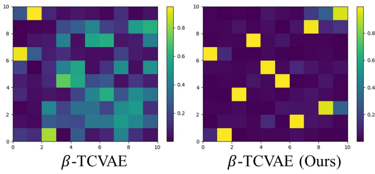


Figure 5: Heatmaps of Eigenvectors for latent vector representations.

Impact of Factor-Aligned Symmetry Size We set the codebook size as 100, and 10 to validate the robustness of our method. In Table 4b, the larger size shows better results than the smaller one and is more stable by showing a low standard deviation.

6.2 Qualitative Analysis Results and Discussion

Is Latent Vector Space Close to Disentangled Space? The previous result as shown in Fig. 1 is a clear example of of whether the latent vector space closely approximates a disentangled space. The latent vector space of previous works (Fig. 1a-1e) are far from disentangled space (Fig. 1g) but CFASL shows the closest disentangled space compare to other methods.

Alignment of Latent Dimensions to Factor-Aligned Symmetry In the principal component analysis of latent vectors shown in Fig. 5, the eigenvectors $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_{|D|}]$ are close to one-hot vectors compared to the baseline, and the dominating dimension of the one-hot vectors are all different. This result implies that the representation (factor) changes are aligned to latent dimensions.

Factor Aligned Symmetries To verify the representation of learnable codebook over composite symmetries and factor-aligned symmetries, we randomly select a sample pair as shown in Fig. 6. The results imply that g_c generated from the codebook represents the composite symmetries between two images (① and ②)

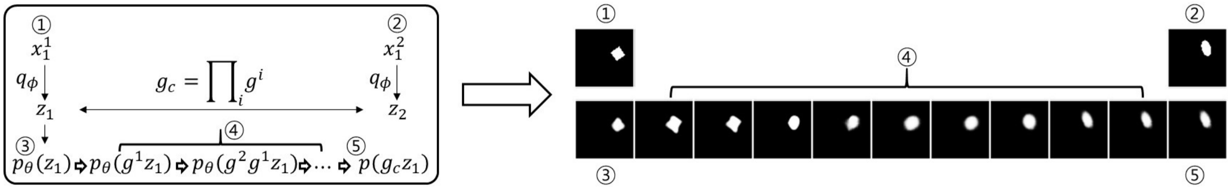


Figure 6: Generated images by composite symmetry and its factor-aligned symmetries. Image ① and ② are inputs, and image ③ is an output from image ① ($p_\theta(z_1)$). Image ⑤ is a output of group element g_c acted on z_1 ($p(g_c z_1)$). Images ④ are outputs of decomposed composite symmetry g_c acted on z_1 sequentially.

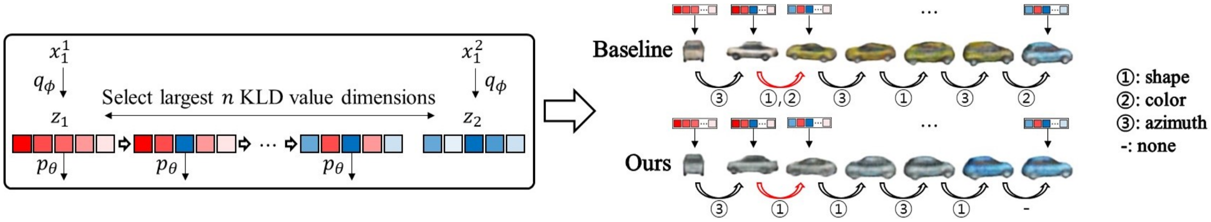


Figure 7: Generated images by dimension change. Red and blue colored squares represent the value of latent vector dimensions of z_1 and z_2. The images of baseline and CAFSL are the generated images from each latent vector.

because the image ② and the generated image ⑤ by symmetry g_c are similar ($p(z_2) \approx p(g_c z_1)$). Also, each factor-aligned symmetry (g_i) generated from codebook section affects a single factor changes as shown in images ④. in Fig. 6.

Factor Aligned Latent Dimension To analyze each factor changes aligned to each dimension of latent vector space, we set the qualitative analysis as shown in 7. We select two random samples (x_1, x_2), generate latent vectors z_1 and z_2, and select the largest Kullback-Leibler divergence (KLD) value dimension from their posterior. Then, replacing the dimension value of z_1 to the value of z_2 one by one sequentially. As a result, the shape and color factors are changed when a single dimension value is replaced within the baseline. However, our method results show no overlapped factor changes compared to baseline results. It implies that each latent vector dimension of the proposed method contains a single factor of information.

Unseen Change Prediction in Sequential Data The sequential observation as Miyato et al. (2022) is rarely observed in our methods, because of the random pairing during training (less 1 pair of observation). But their generated images via trained symmetries of our method are similar to target images as shown in Fig. 8. This result implies that our method is strongly regularized for unseen change.

7 Conclusion

This work tackles the difficulty of disentanglement learning of VAEs in unknown factors change conditions. We propose a novel framework to learn composite symmetries from explicit factor-aligned symmetries by

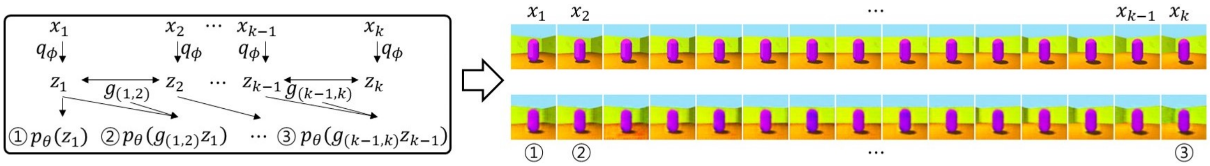


Figure 8: Generalization over unseen pairs of images. We set pairs $\{(x_{i-1}, x_i) | 1 \leq i \leq ||\mathbb{X}|| - 1\}$ then extract the symmetries between elements of each pair $g_p = \{g_{(1,2)}, g_{(2,3)}, \ldots g_{(k-1,k)}\}$ in inference step, where $g_{(k-1,k)}$ is a symmetry between z_{k-1} and z_k. The first row images are inputs (targets) and the second row images are the generated images by symmetry codebook.

codebook to directly represent the multi-factor change of a pair of samples in unsupervised learning. The framework enhances disentanglement by learning an explicit symmetry codebook, injecting three inductive biases on the symmetries aligned to unknown factors, and inducing a group equivariant VAE model. We quantitatively evaluate disentanglement in the condition by a novel metric (m-FVM$_k$) extended from a common metric for a single factor change condition. This method significantly improved in the multi-factor change and gradually improved in the single factor change condition compared to state-of-the-art disentanglement methods of VAEs. Also, training process does not need the knowledge of factor information of datasets. This work can be easily plugged into VAEs and extends disentanglement to more general factor conditions of complex datasets.

References

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: a review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798—1828, August 2013. ISSN 0162-8828. doi: 10.1109/tpami.2013.50. URL `https://doi.org/10.1109/TPAMI.2013.50`.

Diane Bouchacourt, Mark Ibrahim, and Stéphane Deny. Addressing the topological defects of disentanglement via distributed operators, 2021.

Chris Burgess and Hyunjik Kim. 3d shapes dataset. https://github.com/deepmind/3dshapes-dataset/, 2018.

Jinkun Cao, Ruiqian Nai, Qing Yang, Jialei Huang, and Yang Gao. An empirical study on disentanglement of negative-free contrastive learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=fJguu0okUY1`.

Ricky T. Q. Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL `https://proceedings.neurips.cc/paper/2018/file/1ee3dfcd8a0645a25a35977997223d22-Paper.pdf`.

Nima Dehmamy, Robin Walters, Yanchen Liu, Dashun Wang, and Rose Yu. Automatic symmetry discovery with lie algebra convolutional network. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL `https://openreview.net/forum?id=NPOWF_ZLfC5`.

Cian Eastwood and Christopher K. I. Williams. A framework for the quantitative evaluation of disentangled representations. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=By-7dz-AZ`.

Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. *arXiv preprint arXiv:2002.12880*, 2020.

B. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction.* Graduate Texts in Mathematics. Springer International Publishing, 2015. ISBN 9783319134673. URL `https://books.google.co.kr/books?id=didACQAAQBAJ`.

Irina Higgins, Loïc Matthey, Arka Pal, Christopher P. Burgess, Xavier Glorot, Matthew M. Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *ICLR*, 2017.

Irina Higgins, David Amos, David Pfau, Sébastien Racanière, Loïc Matthey, Danilo J. Rezende, and Alexander Lerchner. Towards a definition of disentangled representations. *CoRR*, abs/1812.02230, 2018. URL `http://arxiv.org/abs/1812.02230`.

Irina Higgins, Sébastien Racanière, and Danilo Rezende. Symmetry-based representations for artificial and biological general intelligence, 2022. URL `https://arxiv.org/abs/2203.09250`.

Yeonwoo Jeong and Hyun Oh Song. Learning discrete and continuous factors of data via alternating disentanglement. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 3091–3099. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/jeong19d.html`.

T. Anderson Keller and Max Welling. Topographic vaes learn equivariant capsules. *CoRR*, abs/2109.01394, 2021a. URL `https://arxiv.org/abs/2109.01394`.

T. Anderson Keller and Max Welling. Topographic VAEs learn equivariant capsules. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021b. URL `https://openreview.net/forum?id=AVWROGUWpu`.

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2649–2658. PMLR, 10–15 Jul 2018. URL `https://proceedings.mlr.press/v80/kim18b.html`.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013. URL `https://arxiv.org/abs/1312.6114`.

Abhishek Kumar, Prasanna Sattigeri, and Avinash Balakrishnan. VARIATIONAL INFERENCE OF DISENTANGLED LATENT CONCEPTS FROM UNLABELED OBSERVATIONS. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=H1kG7GZAW`.

Yann LeCun, Fu Jie Huang, and Léon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, CVPR'04, pp. 97–104, USA, 2004. IEEE Computer Society.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3730–3738, 2015.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4114–4124. PMLR, 09–15 Jun 2019. URL `https://proceedings.mlr.press/v97/locatello19a.html`.

Giovanni Luca Marchetti, Gustaf Tegnér, Anastasiia Varava, and Danica Kragic. Equivariant representation learning via class-pose decomposition. In Francisco Ruiz, Jennifer Dy, and Jan-Willem van de Meent (eds.), *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 4745–4756. PMLR, 25–27 Apr 2023. URL `https://proceedings.mlr.press/v206/marchetti23b.html`.

Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dsprites: Disentanglement testing sprites dataset. https://github.com/deepmind/dsprites-dataset/, 2017.

Giangiacomo Mercatali, Andre Freitas, and Vikas Garg. Symmetry-induced disentanglement on graphs. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 31497–31511. Curran Associates, Inc., 2022. URL `https://proceedings.neurips.cc/paper_files/paper/2022/file/cc721384c26c0bdff3ec31a7de31d8d5-Paper-Conference.pdf`.

Nathan Juraj Michlo. Disent - a modular disentangled representation learning framework for pytorch. Github, 2021. URL `https://github.com/nmichlo/disent`.

Takeru Miyato, Masanori Koyama, and Kenji Fukumizu. Unsupervised learning of equivariant structure from sequences. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=7b7iGkuVqlZ`.

Alireza Nasiri and Tristan Bepler. Unsupervised object representation learning using translation and rotation group equivariant VAE. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=qmm__jMjMlL`.

Robin Quessard, Thomas Barrett, and William Clements. Learning disentangled representations and group structure of dynamical environments. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 19727–19737. Curran Associates, Inc., 2020. URL `https://proceedings.neurips.cc/paper_files/paper/2020/file/e449b9317dad920c0dd5ad0a2a2d5e49-Paper.pdf`.

Scott E Reed, Yi Zhang, Yuting Zhang, and Honglak Lee. Deep visual analogy-making. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015. URL `https://proceedings.neurips.cc/paper/2015/file/e07413354875be01a996dc560274708e-Paper.pdf`.

Mehran Shakerinava, Arnab Kumar Mondal, and Siamak Ravanbakhsh. Structuring representations using group invariants. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022. URL `https://openreview.net/forum?id=vWUmBjin_-o`.

Huajie Shao, Shuochao Yao, Dachun Sun, Aston Zhang, Shengzhong Liu, Dongxin Liu, Jun Wang, and Tarek Abdelzaher. ControlVAE: Controllable variational autoencoder. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8655–8664. PMLR, 13–18 Jul 2020. URL `https://proceedings.mlr.press/v119/shao20b.html`.

Huajie Shao, Yifei Yang, Haohong Lin, Longzhong Lin, Yizhuo Chen, Qinmin Yang, and Han Zhao. Rethinking controllable variational autoencoders. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19250–19259, June 2022.

Robin Winter, Marco Bertolini, Tuan Le, Frank Noe, and Djork-Arné Clevert. Unsupervised learning of group invariant and equivariant representations. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022a. URL `https://openreview.net/forum?id=47lpv23LDPr`.

Robin Winter, Marco Bertolini, Tuan Le, Frank Noe, and Djork-Arné Clevert. Unsupervised learning of group invariant and equivariant representations. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022b. URL `https://openreview.net/forum?id=47lpv23LDPr`.

Changyi Xiao and Ligang Liu. Generative flows with matrix exponential. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10452–10461. PMLR, 13–18 Jul 2020. URL `https://proceedings.mlr.press/v119/xiao20a.html`.

Tao Yang, Xuanchi Ren, Yuwang Wang, Wenjun Zeng, and Nanning Zheng. Towards building a group-based unsupervised representation disentanglement framework. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=YgPqNctmyd`.

Xinqi Zhu, Chang Xu, and Dacheng Tao. Commutative lie group VAE for disentanglement learning. *CoRR*, abs/2106.03375, 2021. URL `https://arxiv.org/abs/2106.03375`.

A Loss Function of Baseline

As shown in Table 5, we train the baselines with each objective function.

VAEs	$\mathcal{L}_{VAE}$
β-VAE	$\mathbb{E}_{q_\phi(z\|x)} \log p_\theta(x\|z) - \beta\mathcal{D}_{KL}(q_\phi(z\|x)\|\|p(z))$
β-TCVAE	$\mathbb{E}_{q_\phi(z\|x)} \log p_\theta(x\|z) - \alpha\mathcal{D}_{KL}(q(z,n)\|\|q(z)p(n))$ $-\beta\mathcal{D}_{KL}(q(z)\|\|\prod_j a(z_j)) - \gamma\sum_j \mathcal{D}_{KL}(q(z_j)\|\|p(z_j))$
Factor-VAE	$\frac{1}{N}\sum_i^N [\mathbb{E}_{q(z\|x^i)}[\log p(x^i\|z)] - \mathcal{D}_{KL}(q(z\|x^i)\|\|p(z))]$ $-\gamma\mathcal{D}_{KL}(q(z)\|\|\prod_j(z_j))$
Control-VAE	$\mathbb{E}_{q_\phi(z\|x)} \log p_\theta(x\|z) - \beta(t)\mathcal{D}_{KL}(q_\phi(z\|x)\|\|p(z))$
CLG-VAE	$\mathbb{E}_{a(z\|x)q(t\|z)} \log p(x\|z)p(z\|t)$ $-\mathbb{E}_{q(z\|x)}\mathcal{D}_{KL}(q(t\|z)\|\|p(t)) - \mathbb{E}_{q(z\|x)} \log q(z\|x)$

Table 5: Objective Function of the VAEs.

B Perpendicular and Parallel Loss Relationship

We define parallel loss $\mathcal{L}_p$ to set two vectors in the same section of the symmetries codebook to be parallel: $z - g^i_j \parallel z - g^i_{j'} z$ then,

$$\begin{aligned} & \boldsymbol{z} - g^i_j \boldsymbol{z} = c(\boldsymbol{z} - g^i_{j'}\boldsymbol{z}) && (10) \\ \Rightarrow\;\; & (1-c)\boldsymbol{z} = (g^i_j - c g^i_{j'})\boldsymbol{z} && (11) \\ \Rightarrow\;\; & (1-c)\boldsymbol{I} = g^i_j - c g^i_{j'} \text{ or } [(1-c)I + c g^i_{j'} - g^i_j]\boldsymbol{z} = 0, && (12) \end{aligned}$$

where $\boldsymbol{I}$ is an identity matrix and constant $c \in \mathbb{R}$. However, all latent z is not eigenvector of $[(1-c)\boldsymbol{I} + c g^i_{j'} - g^i_j]$. Then, we generally define symmetry as:

$$g^i_{j'} = \frac{1}{c} g^i_j + \frac{c-1}{c}\boldsymbol{I}, \tag{13}$$

where i, j, and j' are natural number $1 \leq i \leq |S|$, $1 \leq j, j' \leq |SS|$, and $k \neq j$. Therefore, all symmetries in the same section are parallel then, any symmetry in the same section is defined by a specific symmetry in the same section.

We define orthogonal loss $\mathcal{L}_o$ between two vectors, which are in different sections, to be orthogonal: $\boldsymbol{z} - g^i_j\boldsymbol{z} \perp \boldsymbol{z} - g^k_l\boldsymbol{z}$, where $i \neq k$, $1 \leq i, k \leq |S|$, and $1 \leq j, l \leq |SS|$. By the Equation 13,

$$\begin{aligned} & \boldsymbol{z} - g^i_j\boldsymbol{z} \perp \boldsymbol{z} - g^k_l\boldsymbol{z} && (14) \\ \Rightarrow\;\; & (\frac{1}{c_a} g^i_a + \frac{c_a - 1}{c_a}\boldsymbol{I})\boldsymbol{z} - \boldsymbol{z} \perp (\frac{1}{c_b} g^k_b + \frac{c_b - 1}{c_b}\boldsymbol{I})\boldsymbol{z} - \boldsymbol{z} && (15) \\ \Rightarrow\;\; & \frac{1}{c_a}(g^i_a\boldsymbol{z} - \boldsymbol{z}) \perp \frac{1}{c_b}(g^k_b\boldsymbol{z} - \boldsymbol{z}), && (16) \end{aligned}$$

where c_a and c_b are any natural number, and $1 \leq a, b \leq |SS|$. Therefore, if two vectors from different sections are orthogonal and satisfied with Equation 13, then any pair of vectors from different sections is always orthogonal.

C Additional Experiment

C.1 Disentanglement Performance

Reconstruction vs. FVM We conduct the trade-off between the reconstruction and disentanglement performance as shown in Fig. 9. We consider the results on the complex dataset because the trade-off is more distinct in the complex dataset such as 3DShapes.

Complex Dataset We show the model performance with MPI3D dataset.

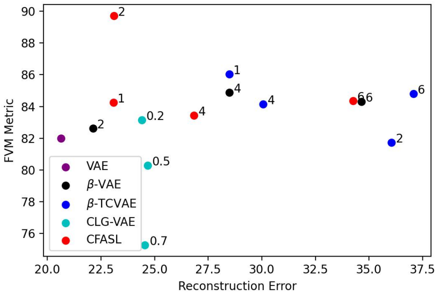


Figure 9: Reconstruction loss vs. Factor VAE metric on 3D Shapes dataset. The numbers next to each plot represent the value of $loss_{\text{rec group}}$ of CLG-VAE and the others are the value of β parameter.

	FVM	beta VAE
β-VAE	16.79(±0.80)	36.40(±8.53)
β-TCVAE	16.95(± 1.38)	51.75(±9.22)
CFASL	**17.44**(±3.33)	**54.80**(±4.54)

Table 6: Disentanglement performance on the MPI3D dataset.

Statistically Significant Improvements As shown in Table 8, our model significantly improves disentanglement learning.

3D Shapes As shown in Table 9, CFASL also shows an advantage on multi-factor change.

C.2 Ablation Studies

How Commutative Lie Group Improves Disetanglement Learning? The Lie group is not commutative, however most factors of the used datasets are commutative. For example, 3D Shapes dataset factors consist of the azimuth (x-axis), yaw (z-axis), coloring, scale, and shape. Their 3D rotations are all commutative. Also, other composite symmetries as coloring and scale are commutative. Even though we restrict the Lie group to be commutative, our model shows better results than baselines as shown in Table 1.

Impact of Commutative Loss on Computational Complexity As shown in Table 7, our methods reduce the composite symmetries computation. Matrix exponential is based on the Taylor series and it needs high computation cost though its approximation is lighter than the Taylor series. We need one matrix exponential computation for composite symmetries with commutative loss, in contrast, the other case needs the number of symmetry codebook elements $|S| \cdot |SS|$ for the matrix exponential and also $|S| \cdot |SS| - 1$ time matrix multiplication.

3D Cars	$\mathcal{L}_c$	without $\mathcal{L}_c$
	x**4.63**	x1.00

Table 7: Complexity.

Comparison of Plug-in Methods To compare plug-in methods, we evaluate common disentanglement metrics on G-VAE Shakerinava et al. (2022) and apply both methods to β-TCVAE. As shown in Table 10, our method shows statistically significant improvements in disentanglement learning although β hyper-parameter of CFASL is smaller than G-VAE. As shown in Table 8, we estimate the p-value over common disentanglement metrics on each dataset. Most values show that improvements in disentanglement learning are statistically significant.

p-value	FVM	MIG	SAP	DCI
dSprites	**0.011**	**0.005**	**0.016**	**0.001**
3D Cars	**0.006**	**0.000**	0.97	**0.003**
smallNORB	**0.000**	**0.002**	**0.000**	1.000

Table 8: p-value estimation on each datasets.

3D Shapes	β-VAE	β-TCVAE	Factor-VAE	Control-VAE	CLG-VAE	OURS
m-FVM$_5$	80.26(±3.78)	79.21(±5.87)	76.69(±5.08)	73.31(±6.54)	73.61(±4.22)	**83.03**(±2.73)

Table 9: m-FVMs results.

Datasets	FVM		MIG		SAP		DCI	
	G-VAE	CFASL	G-VAE	CFASL	G-VAE	CFASL	G-VAE	CFASL
dSprites	69.75(±13.66)	**82.30**(±5.64)	21.09(±9.20)	**33.62**(±8.18)	5.45(±2.25)	**7.28**(±0.63)	31.08(±10.87)	**46.52**(±6.18)
3D Car	92.34(±2.96)	**95.70**(±1.90)	11.95(±2.16)	**18.58**(±1.24)	**2.10**(±0.96)	1.43(±0.18)	26.91(±6.24)	**34.81**(±3.85)
smallNROB	46.64(±1.45)	**61.15**(±4.23)	20.66(±1.22)	**22.23**(±0.48)	10.37(±0.51)	**11.12**(±0.48)	**27.77**(±0.68)	24.59(±0.51)

Table 10: Comparison of disentanglement scores of plug-in methods in single factor change.

C.3 Additional Qualitative Analysis (Baseline vs. CFASL)

Fig. 10-11 show the qualitative results on 3D Cars introduced in Fig. **??**. Fig. 12-13, and Fig. 16 show the dSprites and smallNORB dataset results respectively. Additionally, we describe Fig. 14-15results over 3D Shapes datasets respectively. We randomly sample the images in all cases.

3D Cars As shown in Figure 10c, CFASL shows better results than the baseline. In the 1^{st} and 2^{nd} rows, the baseline changes shape and color factor when a single dimension value is changed, but ours clearly disentangle the representations. Also in the 3^{rd} row, the baseline struggles with separating color and azimuth but CFASL successfully separates the color and azimuth factors.

- 1^{st} row: our model disentangles the *shape* and *color* factors when the 2^{nd} dimension value is changed.
- 2^{nd} row: ours disentangles *shape* and *color* factors when the 1^{st} dimension value is changed.
- 4^{th} row: ours disentangles the *color*, and *azimuth* factors when the 2^{nd} dimension value is changed.

dSprites As shown in Figure 12c, the CFASL shows better results than the baseline. The CFASL significantly improves the disentanglement learning as shown in the 4^{th} and 5^{th} rows. The baseline shows the multi-factor changes during a single dimension value is changed, while ours disentangles all factors.

- 1^{st} row: ours disentangles *the x- and y-pos* factor when the 2^{nd} dimension value is changed.
- 2^{nd} row: ours disentangles the *rotation* and *scale* factor when the 2^{nd} dimension value is changed.
- 3^{rd} row: ours disentangles the *x- and y-pos*, and *rotation* factor when the 1^{st} and 2^{nd} dimension values are changed.
- 4^{th} row: ours disentangles the *all factors* when the 1^{st} and 2^{nd} dimension values are changed.

3D Shapes As shown in Figure 14c, the CFASL shows better results than the baseline. In the 1^{st}, 3^{rd}, and 5^{th} rows, our model clearly disentangles the factors while the baseline struggles with disentangling multi-factors. Even though our model does not clearly disentangle the factors, compared to the baseline, which is too poor for disentanglement learning, ours improves the performance.

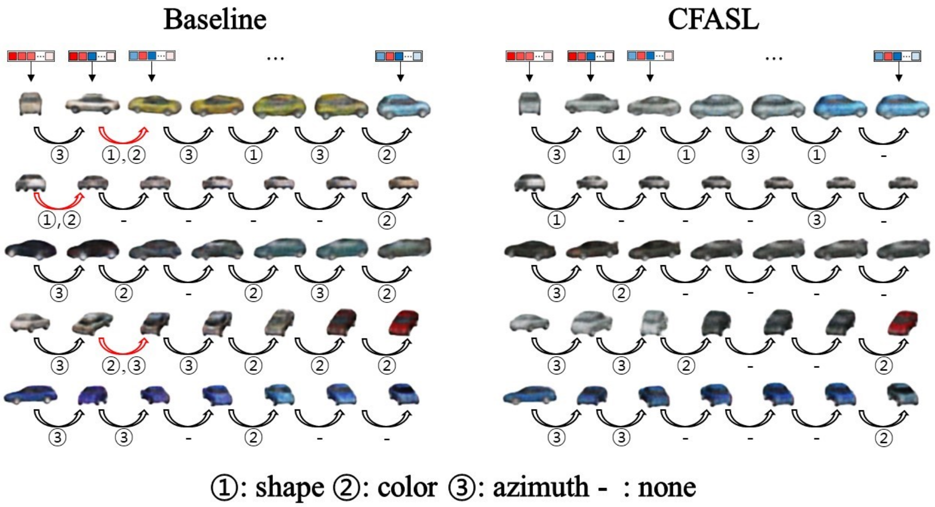
- 1^{st} row: our model disentangles the *object color* and *floor color* factor when the 2^{nd} and 3^{rd} dimension values are changed.
- 2^{nd} row: ours disentangles *shape* factor in 1^{st} dimension, and *object color* and *floor color* factors at the 4^{th} dimension value are changed.
- 3^{rd} row: ours disentangles the *object color* and *floor color* factor when the 3^{rd} dimension value is changed.
- 4^{th} row: ours disentangles the *scale*, *object color*, *wall color*, and *floor color* factor when the 2^{nd} and 3^{rd} dimension values are changed.
- 5^{th} row: ours disentangles the *shape*, *object color*, and *floor color* factor when the 1^{st} and 2^{nd} dimension values are changed.

smallNORB Even though our model does not clearly disentangle the multi-factor changes, ours shows better results than the baseline as shown in Figure 16c.

- 1^{st} row: our model disentangles the *category* and *light* factor when the 2^{nd} dimension value is changed.
- 3^{rd} row: ours disentangles *category* factor and *azimuth* factors when the 5^{th} dimension value is changed.

(a) Generated images by composite symmetry on 3D Cars dataset. The images in the red box are inputs. The images in the blue box at odd column are same as ③ and even column are same as ⑤ in Fig. 6.

(b) Generated images by its factor-aligned symmetries on 3D Cars dataset. The images are same as ④ in Fig 6.



(c) Generated images by dimension change on 3D Cars dataset.

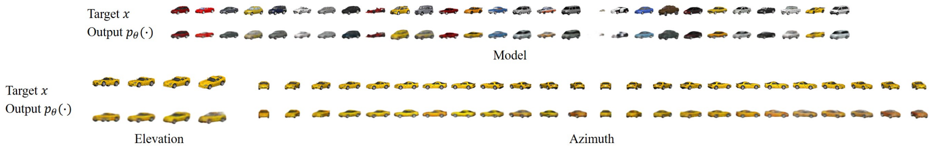
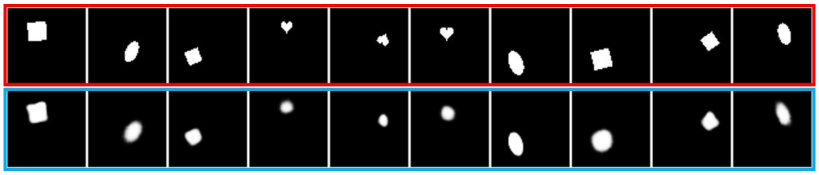
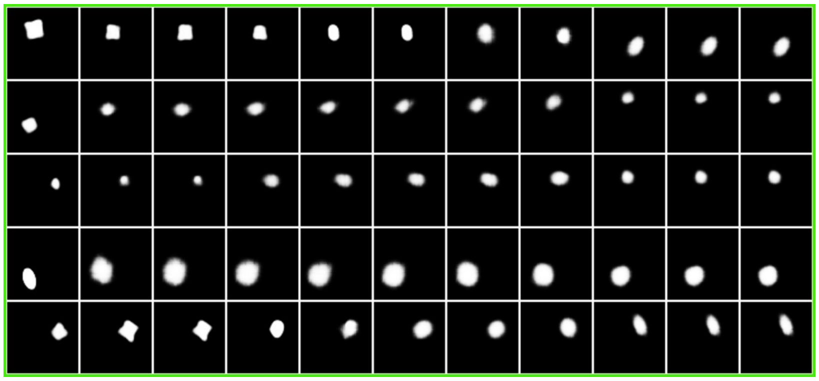


Figure 10: Generalization over unseen pairs of images on 3D Cars dataset.

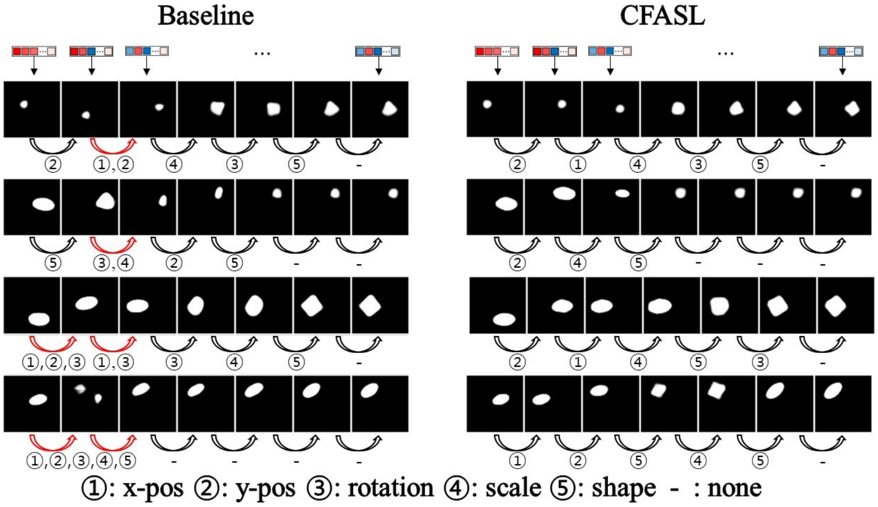
Figure 11: Fig. 10a shows the generation quality of composite symmetries results, Fig. 10b shows the disentanglement of symmetries by factors results, and Fig. 10c shows the disentanglement of latent dimensions by factors results.



(a) Generated images by composite symmetry on dSprites dataset. The images in the red box are inputs. The images in the blue box at odd column are same as ③ and even column are same as ⑤ in Fig. 6.



(b) Generated images by its factor-aligned symmetries on dSprites datset. The images are same as ④ in Fig 6.



(c) Generated images by dimension change on dSprites dataset.

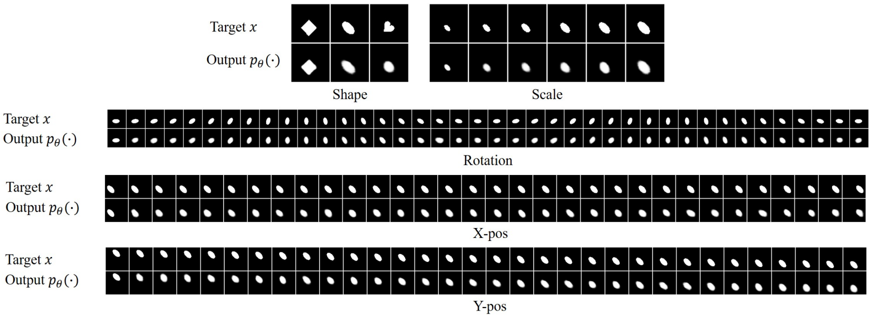
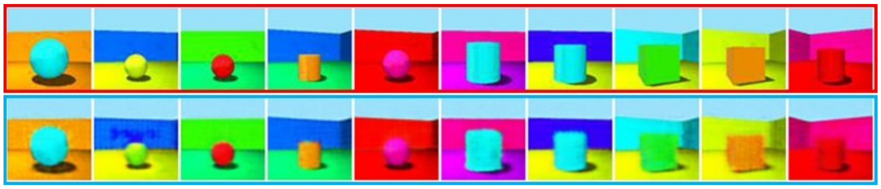
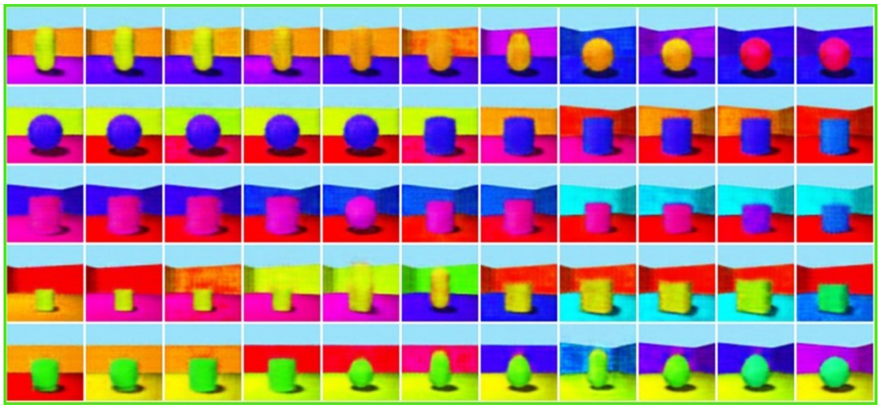


Figure 12: Generalization over unseen pairs of images on dSprites dataset.

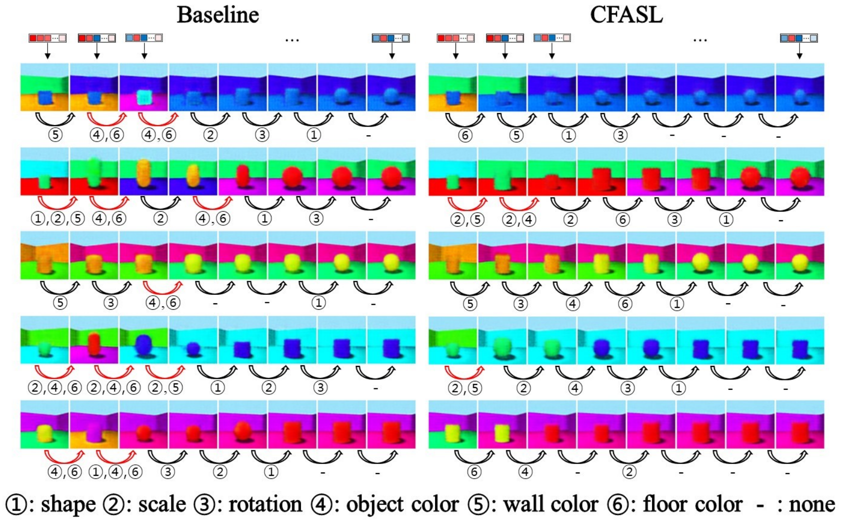
Figure 13: Fig. 12a shows the generation quality of composite symmetries results, Fig. 12b shows the disentanglement of symmetries by factors results, and Fig. 12c shows the disentanglement of latent dimensions by factors results.



(a) Generated images by composite symmetry on 3DShapes dataset. The images in the red box are inputs. The images in the blue box at odd column are same as ③ and even column are same as ⑤ in Fig. 6.



(b) Generated images by its factor-aligned symmetries on 3DShapes datset. The images are same as ④ in Fig 6



(c) Generated images by dimension change on 3DShapes dataset.

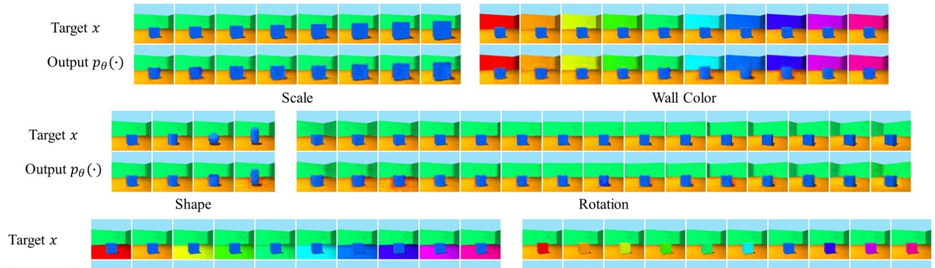
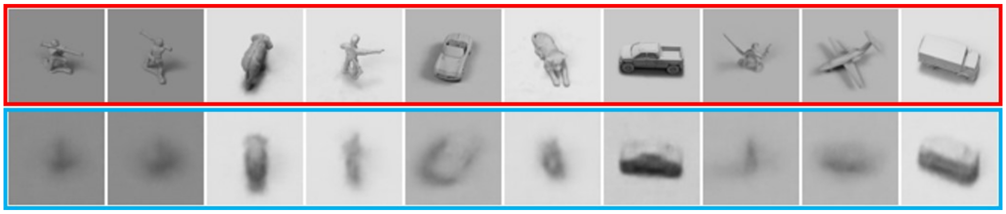
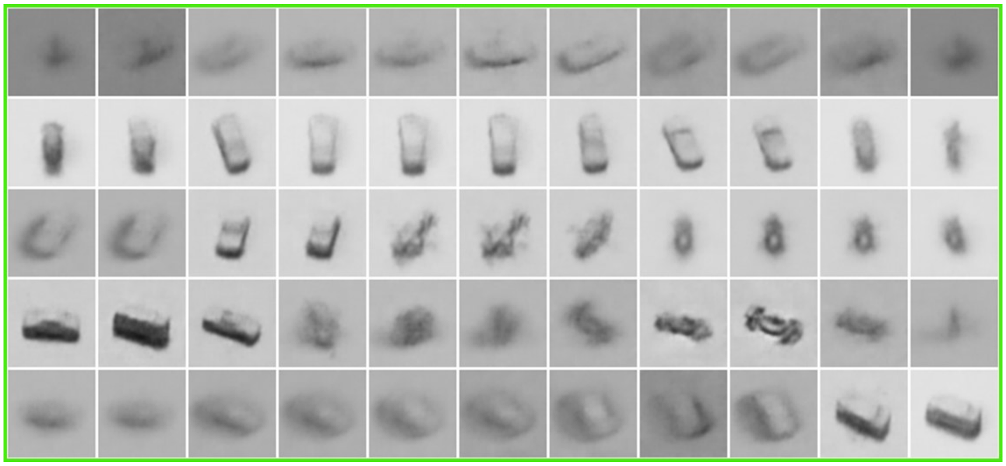


Figure 14: Generalization over unseen pairs of images on 3DShapes dataset.

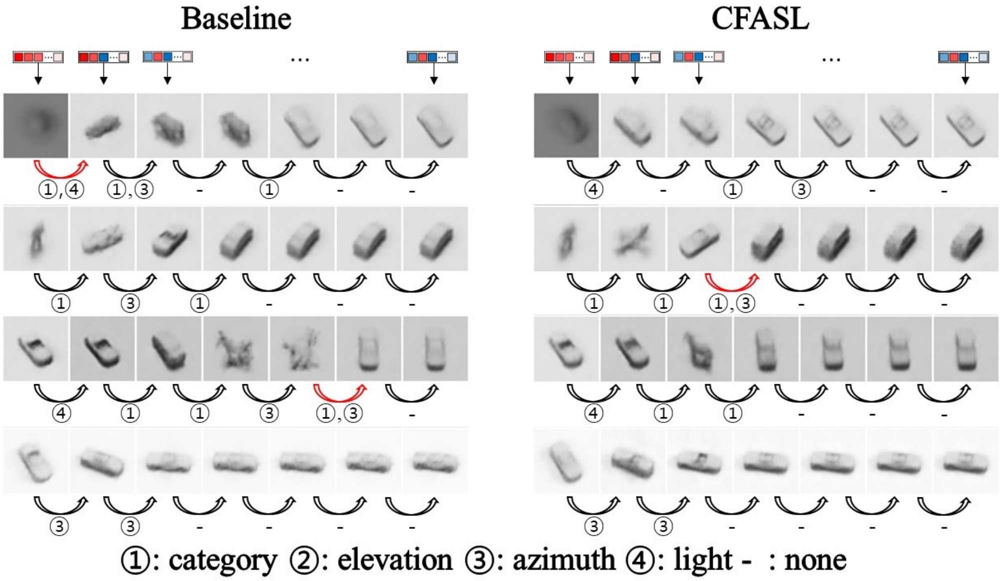
Figure 15: Fig. 14a shows the generation quality of composite symmetries results, Fig. 14b shows the disentanglement of symmetries by factors results, and Fig. 14c shows the disentanglement of latent dimensions by factors results.



(a) Generated images by composite symmetry on smallNORB dataset. The images in the red box are inputs. The images in the blue box at odd column are same as ③ and even column are same as ⑤ in Fig. 6.



(b) Generated images by its factor-aligned symmetries on smallNORB datset. The images are same as ④ in Fig 6



(c) Generated images by dimension change on smallNORB dataset.

Figure 16: Fig. 16a shows the generation quality of composite symmetries results, Fig. 16b shows the disentanglement of symmetries by factors results, and Fig. 16c shows the disentanglement of latent dimensions by factors results.