HIERARCHICAL DISENTANGLE NETWORK FOR OBJECT REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

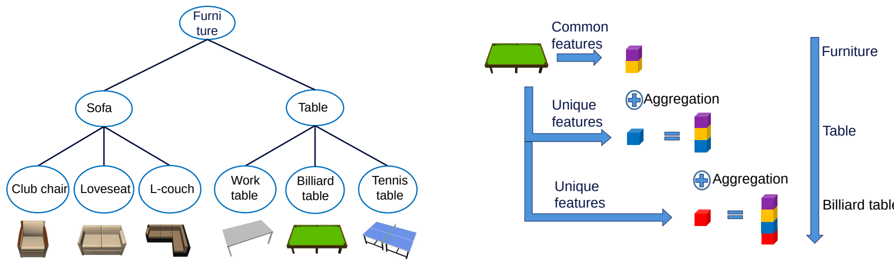
An object can be described as the combination of primary visual attributes. Disentangling such underlying primitives is the long-term objective of representation learning. It is observed that categories have the natural multi-granularity or hierarchical characteristics, i.e. any two objects can share some common primitives in a particular category granularity while they may possess their unique ones in another granularity. However, previous works usually operate in a flat manner (i.e. in a particular granularity) to disentangle the representations of objects. Though they may obtain the primitives to constitute objects as the categories in that granularity, their results are obviously not efficient and complete. In this paper, we propose the hierarchical disentangle network (HDN) to exploit the rich hierarchical characteristics among categories to divide the disentangling process in a coarse-to-fine manner, such that each level only focuses on learning the specific representations in its granularity and finally the common and unique representations in all granularities jointly constitute the raw object. Specifically, HDN is designed based on an encoder-decoder architecture. To simultaneously ensure the disentanglement and interpretability of the encoded representations, a novel hierarchical generative adversarial network (GAN) is elaborately designed. Quantitative and qualitative evaluations on four object datasets validate the effectiveness of our method.

1 INTRODUCTION

Representation learning, as one basic and hot topic in machine learning and computer vision community, has achieved significant progress in recent years on different tasks such as recognition Russakovsky et al. (2015), detection Ren et al. (2015); Redmon et al. (2016); Liu et al. (2016b) and generation Goodfellow et al. (2014), benefiting from the rapid development of representation learned by deep neural networks. Considering the strong capacity of deep representation, in this paper, we mainly focus on the deep representation learning framework.

Despite great success the deep representations have achieved as mentioned above, two important problems are still unresolved or less considered, i.e. the interpretability and the disentanglement of the learned representations. In the past decades, various works have been developed to reveal the black box of deep learning Zeiler & Fergus (2014); Dosovitskiy & Brox (2016b); Bau et al. (2017); Simonyan et al. (2013); Stock & Cissé (2017); Zhang et al. (2017) and move us closer to the goal of disentangling the variations within data Reed et al. (2014); Mathieu et al. (2016); Rifai et al. (2012); Tran et al. (2017); Gonzalez-Garcia et al. (2018); Huang et al. (2018); Chen et al. (2016). Even though they have brought great insights to us, they still have some limitations. For instance, Chen et al. (2016); Xie et al. (2017); Zhao et al. (2017) learn to disentangle variation factors within each category using generative models, instead of investigating the similarities and differences among categories, leading to poor discriminability. Therefore, the learned representations would not well conform to human perception. Though Gonzalez-Garcia et al. (2018); Huang et al. (2018) try to obtain the domain-invariant and domain-specific knowledge, they can only handle two categories one time, which is not that efficient. In this paper, we attempt to learn disentangled representations in a more natural and efficient manner.

Let us first discuss how we human understand an object. Generally speaking, an object can be regarded as the combination of many semantic attributes. Hundreds of thousands of objects in the world can be clustered and recognized by us human just because we can figure out the common and



(a) Hierarchy structure (b) Hierarchical features extraction

Figure 1: An illustration of a hierarchy structure (a) and extracting the hierarchical features that constitute a leaf-level image (b). In (b), the common features that only contain the information of its being the root category are first extracted. By tracing from the root to leaf, the unique features that contain additional information of its being the finer-grained category are further extracted.

unique attributes of an object compared with others. Besides, a man who never play the billiards can only recognize a *table* in an image, while a sports fan may regard it as a *billiard table*. Both of them are right since categories have natural hierarchy structure. In other words, categories are naturally of multi-granularity. As shown in Fig. 1(a), given six leaf-level categories, they can be organized in a three-level hierarchy considering the common and different features they have. Each child category in the hierarchy is a special case of its parent category since it inherits all features from its parent category and has extra features that are not present in its parent category. From another perspective, each parent category is the abstraction of all its child categories considering it contains the attributes that are present in all its child categories. Then we come back to the task of disentangling representation learning. It aims to learn the representation encoding useful information that can be applied in other tasks (e.g. building classifiers and predictors) Bengio et al. (2013). Taking the multi-granularity nature of categories into account, if we only learn the representations of an object in a flat manner for a specific category granularity as previous works do, it will not be scalable and comprehensive for the machine to be qualified for various tasks in the real world.

Our work aims to exploit the natural hierarchical characteristics among categories to divide the representation learning in a coarse-to-fine manner, such that each level only focuses on learning the specific representations in its granularity. For instance, given a *billiard table* image in Fig. 1(b), it tangles the information of being a *furniture*, a *table* and a *billiard table*. We first extract the features that only contain the information of *furniture* from the image. By tracing from the root to leaf level, more and more information is extracted until we can recognize its belonging categories in all hierarchical levels. By doing so, the disentangled representations are expected to find wide and promising applications. For example, one can transfer the semantics in a specific category level from one object to another while keep information of other levels unchanged. Besides, it would help for the hierarchical image compression task using different levels of the disentangled representations. To achieve the objective of multi-granularity disentangling and simultaneously interpreting the results so that we human can understand, we propose the hierarchical disentangle network (HDN), which draws lessons from hierarchical classification and the recent proposed generative adversarial nets Goodfellow et al. (2014). Extensive experiments are conducted on four popular object datasets to validate the effectiveness of our method.

2 Related Works

Disentangling Deep Representations. The goal of disentangling representation learning is to discover factors of variation within data Bengio et al. (2013). Recent years have witnessed a substantial interest on such research area Tenenbaum & Freeman (1996), including works based on deep learning Reed et al. (2014); Mathieu et al. (2016); Rifai et al. (2012); Wang et al. (2017); Tran et al. (2017); Gonzalez-Garcia et al. (2018); Huang et al. (2018); Chen et al. (2016). Rifai et al. (2012) is probably the earliest to learn disentangled representations using deep networks for the task of emotion recognition. Reed et al. (2014) is based on a higher-order Boltzmann machine and regards each factor variation of the manifold as its sub-manifold. Mathieu et al. (2016) and Chen et al. (2016) leverage the generative adversarial nets (GAN) to learn factors of variation. Recently, cross-domain translation methods Gonzalez-Garcia et al. (2018); Huang et al. (2018) learn the domain-invariant and domain-specific representations. These works ignore the existing natural and inherent hierarchy relationships among categories, with which we can conduct the disentangling in a coarse-to-fine manner such that each level only focuses on learning the specific representations in its granularity.

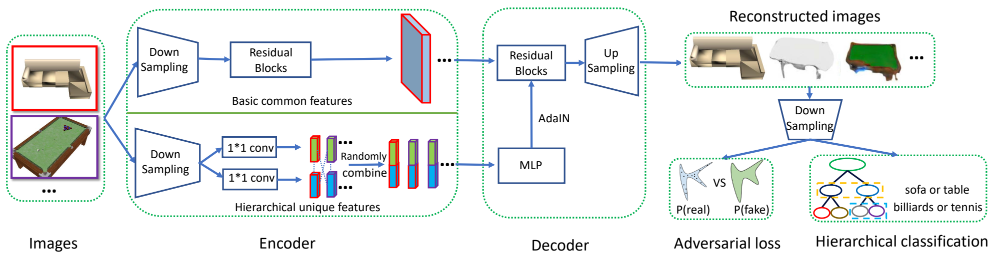


Figure 2: An illustration of the framework of our method. Given images belonging to a hierarchy, the common feature maps of their being the root category and the unique features in different non-root levels that can further distinguish them as finer-grained categories are extracted by the upper residual and bottom 1*1 convolutional branches, respectively. The unique features are transformed as the parameters of Adaptive Instance Normalization (AdaIN) and thus be aggregated into the common feature maps to obtain comprehensive representations of images. To ensure disentanglement of the unique features, different levels of them are randomly combined and then reconstructed in the image space where adversarial loss and hierarchical classification loss are elaborately designed.

Network Interpretability. Network interpretability aims to learn how the network works via visualizing it from the perspective that our human can understand. These methods can be briefly divided into two groups according to whether the visualization is involved in the network during training, i.e. the off-line methods and online methods. The off-line methods make attempts to visualize patterns in image space that activate each convolutional filter Zeiler & Fergus (2014); Dosovitskiy & Brox (2016b;a); Bau et al. (2017; 2019) or to interpret the area in an image that is responsible for the network prediction Simonyan et al. (2013); Fong & Vedaldi (2017); Zintgraf et al. (2017); Abbasi-Asl & Yu (2017); Stock & Cissé (2017); Palacio et al. (2018); Geirhos et al. (2019). While such methods can explain what has already been learned by the model, they cannot improve the model interpretability in return. Instead, the online works propose to directly learn interpretable representations during training Li et al. (2018); Zhang et al. (2017). However, these methods mainly focus on figuring out the running mechanism of networks while paying less attention to dissecting variations of the features among categories, which cannot make models really understand their inputs.

Hierarchy-regularized Learning. Semantic hierarchies have been explored in object classification task for accelerating recognition Griffin & Perona (2008); Marszalek & Schmid (2008), obtaining multiple granularities of predictions Deng et al. (2012); Ordonez et al. (2013), making use of category relation graphs Deng et al. (2014); Ding et al. (2015), and improving recognition performance as additional supervision Zhao et al. (2011); Srivastava & Salakhutdinov (2013); Hwang & Sigal (2014); Yan et al. (2015); Goo et al. (2016); Ahmed et al. (2016). While these discriminative classification works have achieved their expected goals, they usually lack interpretability. To address such issues, Xie et al. (2017); Zhao et al. (2017) propose to use generative models to disentangle the factors from low-level representations to high-level ones that can construct a specific object. Singh et al. (2019) uses an unsupervised generative framework to hierarchically disentangle the background, object shape and appearance from an image. However, they either deal with each category in isolation or ignore the discriminability of learned features, and thus cannot accurately disentangle the differences and similarities among categories.

3 Hierarchical Representation Learning

Supposing that a category hierarchy is given in the form shown in Fig. 1(a), we use $l = 1, ..., L$ to denote the level of hierarchy (L for the leaf level and 1 for the root level), K_l to denote the number of nodes at level l, n_l^k to denote the k-th node at level l, and C_l^k to denote the number of children for n_l^k. As illustrated in Fig. 1(b), given an original object image denoted as $\mathbf{I}^o$, our goal is to extract the feature $\mathbf{F}_l$ in the l-th level.

Generally speaking, an object $\mathbf{O}$ can be described as the combination of a group of visual attributes:

$$\mathbf{O} = \underbrace{\underbrace{\underbrace{\mathbf{A}_1 + ... + \mathbf{A}_i}_{level=1} + \mathbf{A}_{i+1} + ... + \mathbf{A}_j}_{level=2 \quad ...} + \mathbf{A}_{j+1} + ... + \mathbf{A}_m}_{level=l} + \Delta(\mathbf{O}) \tag{1}$$

where $\mathbf{\Delta}(\mathbf{O})$ represents currently undefined attributes existing on $\mathbf{O}$. As we have discussed, we human classify $\mathbf{O}$ to a particular category granularity according to a subset of the whole attribute set in Eqn.(1). Take the object in Fig. 1(b) for example, it can be regarded as a *furniture* since it contains the attribute subset $\{\mathbf{A}_1 + ... + \mathbf{A}_i\}$, and be classified to a *table* in terms of the attribute subset $\{\mathbf{A}_1 + ... + \mathbf{A}_i + \mathbf{A}_{i+1} + ... + \mathbf{A}_j\}$ present in it. Therefore, the disentangled feature $\mathbf{F}_l$ for our objectives in Fig. 1(b) is actually the reflection of the attribute subset formulated in Eqn.(1). Moreover, since the hierarchical correlations (i.e. the inherited relationship) among categories in different hierarchies, obviously the subset $\{\mathbf{A}_1 + ... + \mathbf{A}_i + \mathbf{A}_{i+1} + ... + \mathbf{A}_j\}$ includes the subset $\{\mathbf{A}_1 + ... + \mathbf{A}_i\}$, naturally leading to the disentangled $\mathbf{F}_{l-1}$ being the proper subset of $\mathbf{F}_l$.

Taking these into consideration, we design the hierarchical disentangle network (HDN) based on the autoencoder architecture in Fig. 2. The encoder E dissects the hierarchical representations given a semantic hierarchy. The decoder G plays the role of an interpreter to reflect the variations of semantic in the image space for different hierarchical levels guided by the hierarchical discriminator D_{adv} and classifiers D_{cls} (they share most network architecture except the output layers).

3.1 Top-down learning of Hierarchical Representations

Since $\mathbf{F}_{l-1}$ is the proper subset of $\mathbf{F}_l$, once $\mathbf{F}_{l-1}$ is obtained, only the difference $\mathbf{R}_l$ $(1 < l \leq L)$ between $\mathbf{F}_l$ and $\mathbf{F}_{l-1}$ needs to be encoded. Considering these, we devise a top-down representation extraction scheme.

Given $\mathbf{F}_{l-1}$ and $\mathbf{R}_l$, we aggregate them together to obtain the whole representation in the l-th level. Such procedure can be formulated as:

$$\mathbf{F}_l = \mathbf{F}_{l-1} \oplus \mathbf{R}_l \tag{2}$$

where $\oplus$ means information aggregation. In summary, to hierarchically disentangle an object, the common feature $\mathbf{F}_1$ in the root level and the unique ones $\mathbf{R}_l$ in deeper levels need to be encoded.

To further interpret the semantics of these features to us human, the decoder reconstructs them in the image space. The semantics of $\mathbf{F}_1$ are shared among all its offspring which can be regarded as the invariant content of the object, while that of $\mathbf{R}_l$ is unique for different levels which plays the role of the variant style of the object. Therefore, $\mathbf{F}_1$ and $\mathbf{R}_l$ are processed in the upper and bottom branches respectively to make them play different roles during the reconstruction, as shown in Fig. 2.

3.2 Constraints for the Learning Process

The basic constraints of hierarchical disentanglement are making features in different levels perform their own duties. For an object $\mathbf{O}$, the encoded $\mathbf{F}_1$ and $\mathbf{R}_l$ are complementary, as the constraints of $\mathbf{F}_l$ being the proper subset of $\mathbf{F}_{l+1}$. $\mathbf{F}_1$ should encode just right information to describes its being the root category. Further with $\mathbf{R}_l$, one can distinguish it from its brother categories in the l-th level.

Apart from disentanglement, visualization of features in the image space is also one of our objectives. We turn to the popular conditional generative adversarial nets (cGANs) Mirza & Osindero (2014) which can control generated images given conditions. Our HDN leverages the disentangled features $\mathbf{F}_1$ and $\mathbf{R}_l$ to control the variations of reconstructed images in different category levels.

To ensure $\mathbf{F}_1$ and $\mathbf{R}_l$ be well disentangled, we propose a random combination strategy for different levels of features from different objects and control the generated images through these combined features, as shown in Fig.2. Specifically, given $\mathbf{F}_1^1$, $\mathbf{R}_l^1$ and $\mathbf{F}_1^2$, $\mathbf{R}_l^2$ from arbitrary two objects, we obtain the newly combined features $\mathbf{F}_1$ and $\mathbf{R}_l$, where $\mathbf{F}_1$ and $\mathbf{R}_l$ come from either the first or second object for each level. The newly combined features are aggregated together as the input for the decoder G to generate a new object image $\mathbf{I}^g$. Such image should satisfy the following losses:

– **Hierarchical classification loss**. For each level, $\mathbf{I}^g$ should be classified to the category that $\mathbf{R}_l^i$ reflects (root level $\mathbf{F}_1$ only contains one category), defined as:

$$J_{cls} = \mathbb{E}_{\mathbf{I}^g \sim p(G)}[-\sum_{l=2}^{L}\sum_{c=1}^{C_{l-1}^k} y_l^c log(D_{cls}(\mathbf{I}^g)_l^c)] \tag{3}$$

where J_{cls} is cross-entropy loss among local categories in each level which have a common parent node k such as the dashed rectangled categories in the bottom right corner of Fig.2. $p(G)$ denotes distribution of generated images $G(\mathbf{F}_1^i, \mathbf{R}_l^i)$. $D_{cls}(\mathbf{I}^g)_l^c$ is probabilistic prediction on the c-th local category, and y_l^c is the ground truth local label of the generated object in the l-th level.

Please note that we only focus on the *local* brother categories instead of all categories in that level. It makes the disentanglement more flexible. On one hand, the classificaiton in each level can thus only focus on the unique features that are just discriminative among those local brother categories. On the other hand, the duties of different levels can be well disentangled, since if the semantic information encoded in different levels is tangled, after the random combination and image reconstruction, the hierarchical classifers would be quite confused.

– **Adversarial loss**. We employ GANs to match the distribution of reconstructed images to the real data distribution. Specifically, the LS-GAN Mao et al. (2017) loss is adopted in light of its stable training, defined as:

$$J_{GAN} = \mathbb{E}_{\mathbf{I}^g \sim p(G)} \left[(1 - D_{adv}(\mathbf{I}^g))^2\right] \tag{4}$$

– **Image reconstruction loss**. As for $\mathbf{F}_1$ and $\mathbf{R}_l$ from one same object, we should be able to reconstruct it as close to the input as possible.

$$J_{recon}^{\mathbf{I}} = \mathbb{E}_{\mathbf{I}^r \sim p'(G)} \left[||\mathbf{I}^r - \mathbf{I}^o||_1\right] \tag{5}$$

where $p'(G)$ is the distribution of generated images taking $\mathbf{F}_1, \mathbf{R}_l$ from the same objects as inputs.

– **Feature reconstruction loss**. Apart from the image reconstruction loss, the feature reconstruction loss is added to HDN to stabilize the training process.

$$J_{recon}^{\mathbf{F},\mathbf{R}} = \mathbb{E}_{(\mathbf{F}_1, \mathbf{R}_l) \sim p(E)} \left[||E(G(\mathbf{F}_1, \mathbf{R}_l)) - (\mathbf{F}_1, \mathbf{R}_l)||_1\right] \tag{6}$$

where $p(E)$ is the distribution of encoded hierarchical features $E(\mathbf{I}^o)$.

Now we combine the four loss functions defined in Eqn.(3), Eqn.(4), Eqn.(5) and Eqn.(6) into one comprehensive loss function for supervising the disentangling of E and visualization of G:

$$J(E, G) = J_{cls} + J_{GAN} + \alpha J_{recon}^{\mathbf{I}} + \beta J_{recon}^{\mathbf{F},\mathbf{R}} \tag{7}$$

where α and β are the hyper-parameters to balance the weights of the four terms.

As for the update of discriminator and hierarchical classifiers, we use the following loss:

$$\begin{aligned} J(D) =& (\mathbb{E}_{\mathbf{I}^o \sim p(data)}[-\sum_{l=2}^{L} \sum_{c=1}^{C_{l-1}^k} y_l^c log(D_{cls}(\mathbf{I}^o)_l^c)]) \\ &+ (\mathbb{E}_{\mathbf{I}^o \sim p(data)} \left[(1 - D_{adv}(\mathbf{I}^o))^2\right] + \mathbb{E}_{\mathbf{I}^g \sim p(G)} \left[(D_{adv}(\mathbf{I}^g))^2\right]) \end{aligned} \tag{8}$$

4 EXPERIMENTS

Datasets: We conduct experiments on hierarchical annotated data from four datasets, typical examples in the hierarchy are shown in Fig.8, Fig.9 and Fig.10 in Appendix. The first is CelebA dataset Liu et al. (2015). It provides more than 200K face images with 40 attribute annotations. Following the official train/test protocol, we define a four-level hierarchy structure which has explicit attribute difference between any two levels. Specifically, all faces (root category) are first divided into two categories based on gender. Such initial categories are further classified according to the smile expression and hair color in the next two levels. With such ground-truth hierarchical annotations, we can validate our method more easily.

The second dataset named Fashion-MNIST Xiao et al. (2017) is proposed as a direct drop-in replacement for the original MNIST dataset for benchmarking machine learning algorithms. It shares the same train/test split with MNIST. Since such dataset does not provide any hierarchical structure, we cluster T-shirt, coat, pullover as one super category and trouser, dress as another super one to construct a three-level hierarchy (root is fashion) according to their appearance similarities.

The other two datasets are 3D data, CADCars Fidler et al. (2012) and ShapeNet Chang et al. (2015). CADCars contains 183 3D Car models and ShapeNet is constitutive of 51,300 3D models covering 55 common and 205 finer-grained categories. Using the provided tools, we generated 24 2D images with 6 pose and 4 illumination variations for CADCars. These 2D data are clustered into four super categories, i.e. minibus, sedan, sports and SUV, and are further divided into 6 finer-grained categories for each super one based on pose annotations, which defines a three-level hierarchy. On

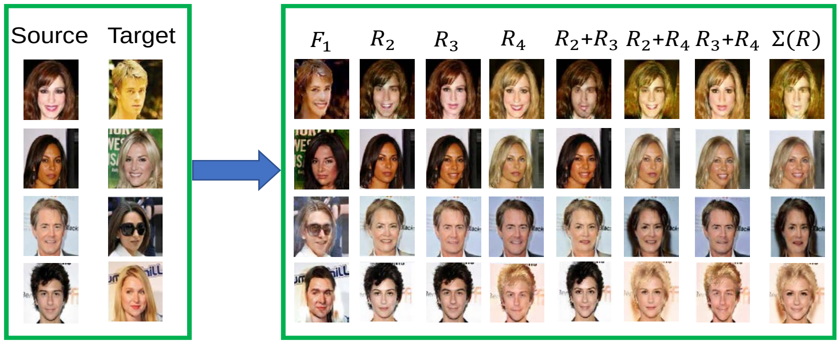
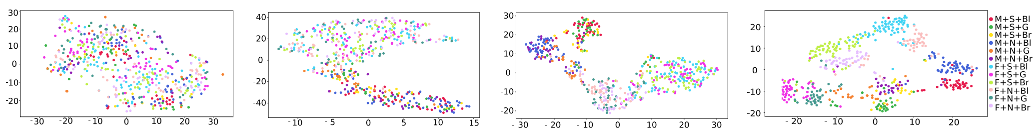


Figure 3: Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on CelebA. Different columns denote results of using $\mathbf{F}_1$, $\mathbf{R}_l$ or their combinations disentangled from the target images to replace the corresponding levels of the sources. Ground truths of $\mathbf{R}_2$, $\mathbf{R}_3$, $\mathbf{R}_4$ are gender, smile and hair color, respectively.



(a) Root (b) Level 2 (c) Level 3 (d) Level 4

Figure 4: 2D tSNE of disentangled $\mathbf{F}_l$ on test set of CelebA for different levels. For easy understand, M and F mean male and female, S and N mean Smile and Neural, Bl, G and Br mean Black, Golden and Brown hair respectively.

ShapeNet, 12 2D images with pose variation are obtained for each 3D model. One three-level category-pose hierarchy named as ShapeNet-P similar to CADCars and one three-level hierarchy named as ShapeNet-C as in Fig.1.(a) are defined. Ratio of train/test split is 4:1.

Implementation Details: Our HDN is implemented with Pytorch platform[1]. Design of the backbone follows recent proposed image generation Karras et al. (2018) and translation works Huang et al. (2018). Images are resized to 128*128 resolution for all datasets except Fashion-MNIST which is resized to 28*28. As shown in Fig.2, to match it with our task, we increase the number of 1*1 convolution branches such that originally one representation is disentangled into multiple hierarchical levels. We also equip the residual blocks with Adaptive Instance Normalization (AdaIN) whose parameters are dynamically generated by a multi-layer perception (MLP) from the disentangled latent codes. Besides, D has L output branches, one for real/fake predictions and the others for hierarchical classifications. More training details are given in the Appendix.

4.1 Disentangled Results

Fig.3 (and Fig.11, Fig.12 in the Appendix) shows the semantics of hierarchically disentangled features. It is observed that different level of features perform their own duties, i.e. they carry just enough information to control the variations within that level (e.g. gender, smile and hair color for CelebA we specially predefined on CelebA), but would not involve more belongs to other level. For instance, in Fig.3 change features of an image in arbitary one level, two levels or all levels to those of another image, the semantics would be changed correspondingly. Apart from the unique $\mathbf{R}_l$, the common feature $\mathbf{F}_1$ also encodes information that is not discriminative among its offspring categories but is necessary to construct the object (e.g. the identity, pose and even the background information of a face image). To give a more intuitional feeling about the ability of disentangled features, we investigate the discriminabilites of them via the popular tSNE tool Maaten & Hinton (2008). As shown in Fig.4 (and Fig.13, Fig.14, Fig.15, Fig.16 in the Appendix), with only the common feature $\mathbf{F}_1$, samples are mixed together. When progressively be combined with features of deeper levels $\mathbf{R}_l$, samples are better separated and almost consistent with the hierarchy structure.

Apart from directly editing an image with the disentangled features, we also show that one can synthesize images that have the visual appearance between the source and target images by interpolation

[1]The source codes will be released to the public.

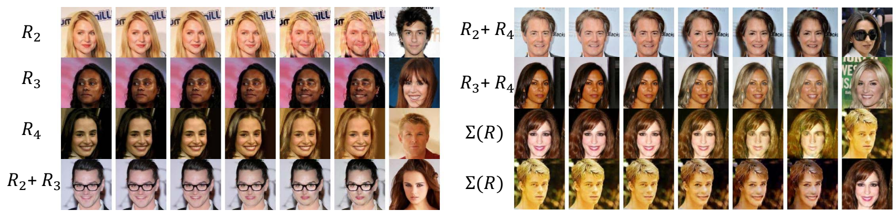


Figure 5: Interpolations of disentangled features between two images (left and right) on CelebA.

	CelebA		Fashion-MNIST		CADCars		ShapeNet-C		ShapeNet-P	
	Test	Gen.	Test	Gen.	Test	Gen.	Test	Gen.	Test	Gen.
Level 2	0.9570	0.9387	0.9629	0.9779	0.9781	0.9792	0.9941	0.9941	0.9323	0.8863
Level 3	0.9232	0.9103	0.9336	0.8464	0.9798	0.9670	0.9844	0.8865	0.9190	0.8413
Level 4	0.8932	0.8799	–	–	–	–	–	–	–	–

Table 1: Accuracy of hierarchical classifications for test and generated (Gen.) images.

in a particular semantic hierarchy level. Such examples are shown in Fig.5. Learning a smooth feature space with continuous variations is an significant issue for representation learning task, which can ensure the generalization ability for unseen similar objects. We have investigated this in Sec.4.3. At the last of this section, a quantitative evaluation of these results is conducted. To be specific, we use the learned hierarchical classifier D to evaluate whether the semantics are correctly disentangled and decoded into the randomly translated images as in Fig.3. To ensure D is reliable, the accuracy of hierarchical classifications on the test data is given as a reference. Table.4.1 gives the evaluation results. Firstly, it can be reached the semantics of translated images with changing different levels are recognized correctly by the corresponding classifiers. Secondly, the deeper of the level, the more difficult of the translation, since the criteria for distinguish one category from others in the deeper level would become more and more complicated (summation of all criteria above this level). Finally, it becomes difficult to transfer the unique features and generate images when that information is difficult to be described and disentangled such as in the leaf-level of Fashion-MNIST and ShapeNet-C, which is deserved to make more efforts.

4.2 Application to Image Retrieval

One of the objectives of learned representations is to applied in real-world applications. Semantic image retrieval has been studied for years. Hashing is one effective and space-time efficient solution for this task. However, the semantic of target images that users expect are not always consistent just due to the tangled information of objects in different hierarchical levels. In this section, we conduct retrieval in different levels on CelebA and compare three competing deep hashing methods: DSH Liu et al. (2016a), HashNet Cao et al. (2017) and SSDH Yang et al. (2018). The backbones of them are same with the bottom branch of encoder E of HDN and pretrained on CASIA WebFace dataset Yi et al. (2014). In l-th level, a model with bit-length as same as the dimension of $\Sigma(\mathbf{R}_l)$ is trained. Images of the test set are used to retrieve the training set.

Table.2 gives the mAP results in different semantic levels. First, our HDN achieves compared performance, though we did not impose specific metric objectives on features. Second, HDN is more efficient since it only needs one model owe to disentanglement while others have to train a model in each level. Third, the retrieval of HDN is more interpretable. As shown in Fig.6, with different parts of features, the returned images satisfy different semantic requirement, while for general method like SSDH one can not interpret the meanings of different code parts.

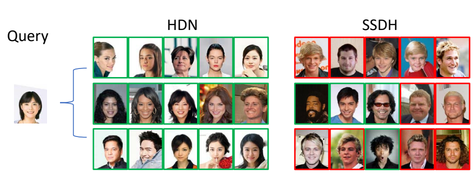


Figure 6: Top-5 returned images of a retrieval case using different parts of features (i.e. different $\mathbf{R}_l$ of HDN, and different bit parts of SSDH). Green and red boxes are correct and false samples judged by the hierarchy.

	DSH	HashNet	SSDH	HDN
Level 2	0.9523	0.9483	0.9593	0.9571
Level 3	0.8010	0.8374	0.8445	0.8589
Level 4	0.6461	0.6336	0.7052	0.6941

Table 2: mAP results of retrieval for compared methods in different semantic levels.

	CelebA		ShapeNet-C		ShapeNet-P	
	Seen	Bald	Seen	Leather couch	Seen	Frontal
Level-2 Acc.	0.9441	0.9850	1.0	1.0	0.8563	0.8125
Level-3 Acc.	0.8520	0.8800	–	–	–	–
Leaf-entropy	0.1779	0.4270	0.1204	0.6472	0.1567	0.9088

Table 3: Hierarchical prediction performance for seen test set and unseen leaf-level categories.

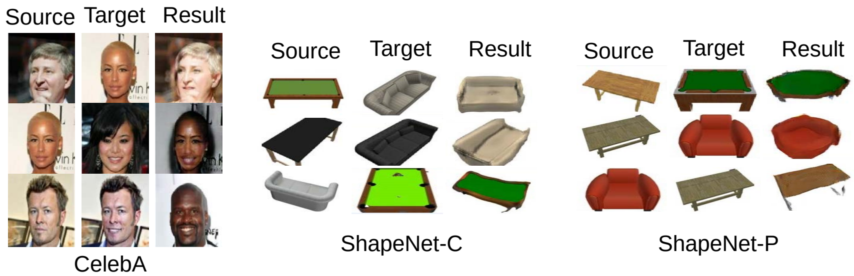


Figure 7: Semantic translation results between seen and unseen objects. Here we replace the source images with all levels of $\mathbf{R}_l$ of the targets, i.e. the right most case in Fig.3.

4.3 Unseen Category Prediction and Semantic Edit

Recognition of unseen categories is a quite challenging task for deep leraning models, which has high requirements for the generalization ability of models. As our HDN learns features in different hierarchy levels, it can obtain sequenced category predictions for an object. Therefore, if unseen objects share similarities with seen ones, one should still obtain the right predictions in those levels, and the predictions among seen categories in levels where unseen objects have their own features should be confused. For levels with similarities and with their own features, accuracy and entropy of predictions of a linear hierarchical classifier trained with the disentangled features are used as evaluation metrics respectively. In this section, we test HDN on certain unseen leaf-level categories, i.e. bald faces of CelebA, leather couch of ShapeNet-C and objects with frontal pose of ShapeNet-P.

Table.3 shows the quantitative results. Two conclusions can be reached. 1). In levels where seen and unseen objects share similarities (i.e. gender, smile levels on CelebA, Sofa/Table level on ShapeNet-C, Loveseat/Club chair/Work table/Billiards on ShapeNet-P), all objects can be correctly classified. 2). In the leaf-level, unseen objects have the unique unseen features, leading to the prediction entropy increase obviously compared with that of seen objects. Besides, it is found that the unseen objects are more likely classified to similar seen categories in leaf-level. For instance, about 30% and 56% bald faces are recognized as black and golden hair respectively, 50% and 50% leather couches are predicted as loveseat and L-couch respectively, 44% and 50% of the frontal sofa/table are classified as the right 30° offset of frontal and left 30° offset of frontal. The semantic translations in Fig.7 between seen and unseen images also verify such results. The semantics of non-leaf levels can be transferred as usual, but the unseen unique features are not. Bald may be disentangled as golden or black hair due to the skin color. The material of leather is ignored in ShapeNet-C since model focus more on shape to distinguish seen objects rather than material during training. The translations to frontal pose are also confused as can be found in the cases of ShapeNet-P. Through this study, we think that disentangling the visual primitives of objects as learned knowledge is one of the most promising solutions for the ability of open-world recognition.

5 Conclusions

We propose the hierarchical disentangle network (HDN) which exploits the natural characteristic-s among categories to divide the representation learning in a coarse-to-fine manner. Our model achieves promising disentangle results. We also show the applications of such disentangled features on semantic translation, retrieval and even unseen objects prediction. However, our work is just an early step towards the long goal of disentangled representation learning, limited by the capacity of generative models on large scale and heavily tangled categories, the performance of HDN is not quite well such as on the ImageNet dataset which deserves to make more efforts.

REFERENCES

Reza Abbasi-Asl and Bin Yu. Interpreting convolutional neural networks through compression. *CoRR*, abs/1711.02329, 2017.

Karim Ahmed, Mohammad Haris Baig, and Lorenzo Torresani. Network of experts for large-scale image categorization. In *ECCV*, pp. 516–532, 2016.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *IEEE,CVPR*, pp. 3319–3327, 2017.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B. Tenenbaum, William T. Freeman, and Antonio Torralba. GAN dissection: Visualizing and understanding generative adversarial networks. In *ICLR*, 2019.

Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, 2013.

Zhangjie Cao, Mingsheng Long, Jianmin Wang, and Philip S. Yu. Hashnet: Deep learning to hash by continuation. In *IEEE , ICCV*, pp. 5609–5618, 2017.

Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *CoRR*, abs/1512.03012, 2015.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *NIPS*, pp. 2172–2180, 2016.

Jia Deng, Jonathan Krause, Alexander C. Berg, and Fei-Fei Li. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *IEEE,CVPR*, pp. 3450–3457, 2012.

Jia Deng, Nan Ding, Yangqing Jia, Andrea Frome, Kevin Murphy, Samy Bengio, Yuan Li, Hartmut Neven, and Hartwig Adam. Large-scale object classification using label relation graphs. In *ECCV*, pp. 48–64, 2014.

Nan Ding, Jia Deng, Kevin P. Murphy, and Hartmut Neven. Probabilistic label relation graphs with ising models. In *IEEE, ICCV*, pp. 1161–1169, 2015.

Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *NIPS*, pp. 658–666, 2016a.

Alexey Dosovitskiy and Thomas Brox. Inverting visual representations with convolutional networks. In *IEEE,CVPR*, pp. 4829–4837, 2016b.

Sanja Fidler, Sven J. Dickinson, and Raquel Urtasun. 3d object detection and viewpoint estimation with a deformable 3d cuboid model. In *NIPS*, pp. 620–628, 2012.

Ruth C. Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *IEEE ,ICCV*, pp. 3449–3457, 2017.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *ICLR*, 2019.

Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. *CoRR*, abs/1805.09730, 2018.

Wonjoon Goo, Juyong Kim, Gunhee Kim, and Sung Ju Hwang. Taxonomy-regularized semantic deep convolutional neural networks. In *ECCV*, pp. 86–101, 2016.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pp. 2672–2680, 2014.

Gregory Griffin and Pietro Perona. Learning and using taxonomies for fast visual categorization. In *IEEE,CVPR*, 2008.

Xun Huang, Ming-Yu Liu, Serge J. Belongie, and Jan Kautz. Multimodal unsupervised image-to-image translation. In *ECCV*, pp. 179–196, 2018.

Sung Ju Hwang and Leonid Sigal. A unified semantic embedding: Relating taxonomies and attributes. In *NIPS*, pp. 271–279, 2014.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *CoRR*, abs/1812.04948, 2018.

Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *AAAI*, 2018.

Haomiao Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Deep supervised hashing for fast image retrieval. In *IEEE,CVPR*, pp. 2064–2072, 2016a.

Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *ECCV*, pp. 21–37, 2016b.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *IEEE ,ICCV*, pp. 3730–3738, 2015.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Xudong Mao, Qing Li, Haoran Xie, Raymond Y. K. Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *IEEE , ICCV*, pp. 2813–2821, 2017.

Marcin Marszalek and Cordelia Schmid. Constructing category hierarchies for visual recognition. In *ECCV*, pp. 479–491, 2008.

Michaël Mathieu, Junbo Jake Zhao, Pablo Sprechmann, Aditya Ramesh, and Yann LeCun. Disentangling factors of variation in deep representation using adversarial training. In *NIPS*, pp. 5041–5049, 2016.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

Vicente Ordonez, Jia Deng, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. From large scale image categorization to entry-level categories. In *IEEE, ICCV*, pp. 2768–2775, 2013.

Sebastian Palacio, Joachim Folz, Jörn Hees, Federico Raue, Damian Borth, and Andreas Dengel. What do deep networks like to see? *CoRR*, abs/1803.08337, 2018.

Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *IEEE,CVPR*, pp. 779–788, 2016.

Scott E. Reed, Kihyuk Sohn, Yuting Zhang, and Honglak Lee. Learning to disentangle factors of variation with manifold interaction. In *ICML*, pp. 1431–1439, 2014.

Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, pp. 91–99, 2015.

Salah Rifai, Yoshua Bengio, Aaron C. Courville, Pascal Vincent, and Mehdi Mirza. Disentangling factors of variation for facial expression recognition. In *ECCV*, pp. 808–822, 2012.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

Krishna Kumar Singh, Utkarsh Ojha, and Yong Jae Lee. Finegan: Unsupervised hierarchical disentanglement for fine-grained object generation and discovery. In *CVPR*, 2019.

Nitish Srivastava and Ruslan Salakhutdinov. Discriminative transfer learning with tree-based priors. In *NIPS*, pp. 2094–2102, 2013.

Pierre Stock and Moustapha Cissé. Convnets and imagenet beyond accuracy: Explanations, bias detection, adversarial examples and model criticism. *CoRR*, abs/1711.11443, 2017.

Joshua B. Tenenbaum and William T. Freeman. Separating style and content. In *NIPS*, pp. 662–668, 1996.

Luan Tran, Xi Yin, and Xiaoming Liu. Disentangled representation learning GAN for pose-invariant face recognition. In *IEEE,CVPR*, pp. 1283–1292, 2017.

Chaoyue Wang, Chaohui Wang, Chang Xu, and Dacheng Tao. Tag disentangled generative adversarial network for object image re-rendering. In *IJCAI*, pp. 2901–2907, 2017.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

Jianwen Xie, Yifei Xu, Erik Nijkamp, Ying Nian Wu, and Song-Chun Zhu. Generative hierarchical learning of sparse FRAME models. In *IEEE, CVPR*, pp. 1933–1941, 2017.

Zhicheng Yan, Hao Zhang, Robinson Piramuthu, Vignesh Jagadeesh, Dennis DeCoste, Wei Di, and Yizhou Yu. HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition. In *IEEE, ICCV*, pp. 2740–2748, 2015.

Huei-Fang Yang, Kevin Lin, and Chu-Song Chen. Supervised learning of semantics-preserving hash via deep convolutional neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(2):437–451, 2018.

Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z. Li. Learning face representation from scratch. *CoRR*, abs/1411.7923, 2014.

Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pp. 818–833, 2014.

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. *CoRR*, abs/1710.00935, 2017.

Bin Zhao, Fei-Fei Li, and Eric P. Xing. Large-scale category structure aware image categorization. In *NIPS*, pp. 1251–1259, 2011.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. Learning hierarchical features from deep generative models. In *ICML*, pp. 4091–4099, 2017.

Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *CoRR*, abs/1702.04595, 2017.

A Appendix

A.1 Network architectures and training details

A.1.1 Network Architectures

Following the backbone designs in Huang et al. (2018) for image-to-image translation task, let *c7s1-k* denotes a 7×7 convolution block with k filters and stride 1. *dk* means a 4×4 convolution block with k filters and stride 2. *Rk* denotes a residual block that contains two 3×3 convolution blocks with k filters. The last layers for different hierarchy levels (common features of root level are encoded by the content encoder) in the style encoder include multiple *c1s1-8* branches, i.e. 1×1 convolution block with 8 filters and stride 1. *uk* denotes a $2\times$ nearest-neighbor upsampling layer followed by a 5×5 convolution block with k filters and stride 1. GAP denotes a global average pooling layer. Instance Normalization (IN) and Adaptive Instance Normalization (AdaIN) are adopted to the content encoder branch and decoder respectively. No normalization is used in the style encoder branch. Use ReLU activations in the encoder-decoder and Leaky ReLU with slope 0.2 in the discriminator and classifier. Multi-scale discriminators with 3 scales (single scale for Fashion-MNIST due to its too small resolutions) are used to ensure both realistic details and global structure. The last layer of the decoder is equipped with a tanh activations to normalize the values of generated images to the range of $[-1, 1]$. In the following, we give the detailed architectures of each module on different datasets.

CelebA, CADCars & ShapeNet:

Content encoder: c7s1-64, d128, d256, R256, R256, R256

Style encoder: c7s1-64, d128, d256, d256, d256, GAP, c1s1-8

Decoder: R256, R256, R256, u128, u64, c7s1-3

Discriminator & Classifier: d64, d128, d256, d512

Fashion-MNIST:

Content encoder: c7s1-32, d64, d128, R128, R128, R128

Style encoder: c7s1-32, d64, d128, R128, R128, R128, GAP, c1s1-8

Decoder: R128, R128, R128, u64, u,32 c7s1-1

Discriminator & Classifier: d32, d64, d128, d256

A.1.2 TRAINING HYPERPARAMETERS

We use the Adam optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.999$, and initial learning rate of 0.0001. We train HDN on all datasets for 300K iterations and half decay the learning rate every 100K iterations. We set batch size to 16. The loss weights α and β in Eqn.(7) are set as 10 and 1 respectively. Random mirroring is applied during training.

A.2 HIERARCHICAL DATA CONSTRUCTION

In this section, Fig.8, Fig.9 and Fig.10 provide leaf-level examples for better understanding the commonalities and individualities among categories in different hierarchy levels. Take the CelebA for example, the root category *face* has two children distinguished by gender attribute. For each of the two super categories, it includes two finer granular children which are further divided by the smile expression. Finally in the leaf-level, each local branch are classified according to their hair colors, i.e. black, golden and brown hair. Within each leaf-level category, samples mainly contain intra-class variations caused by identities, age, pose, etc.

A.3 MORE EXPERIMENTAL RESULTS

In this section, we give disentangled results on CADCars, Fashion-MNIST and ShapeNet. Fig.11 and Fig.12 show the semantics of disentangled features which can well change the variations of generated images. Similarly, Fig.13, Fig.14, Fig.15 and Fig.16 show that progressively involving more features of deeper levels, samples become better separated, which verifies the results of disentanglement are consistent with the semantic hierarchy structures.

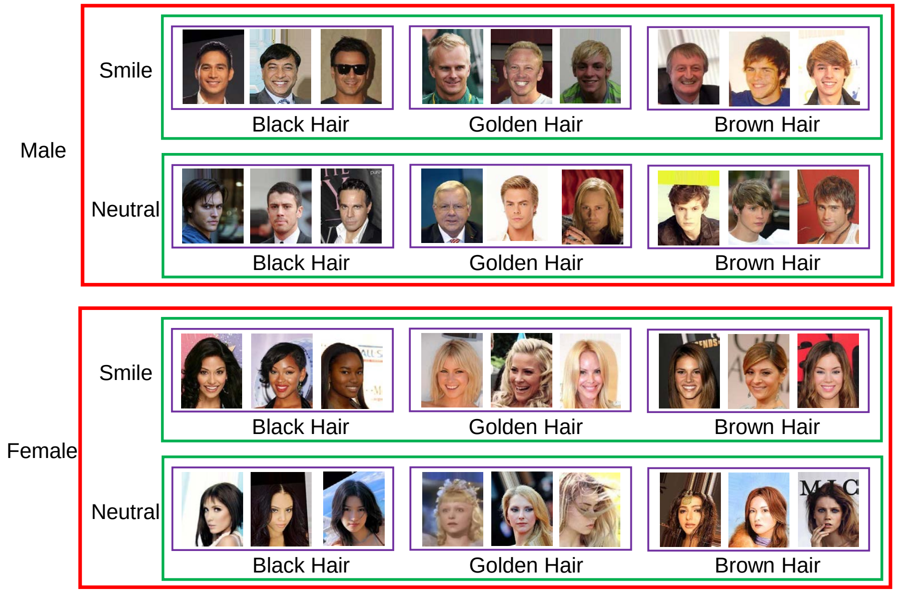
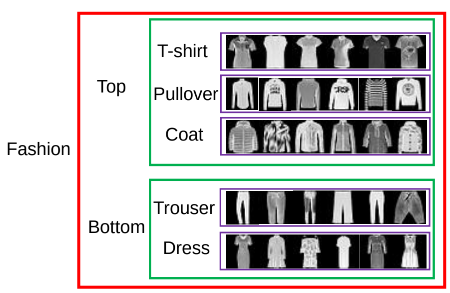
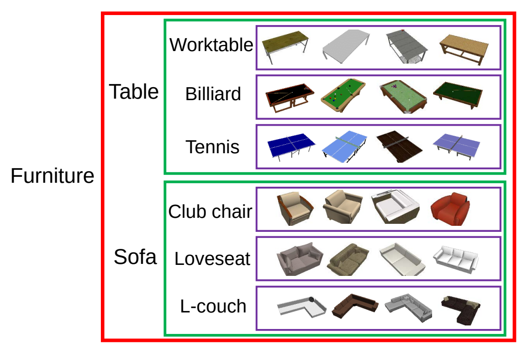


Figure 8: Typical samples of hierarchical data on CelebA. Images within a purple rectangular box are some instances of a leaf-level category. Categories within a green rectangular box belong to one common super-category. The super-categories within a red rectangular box share one common ancestor.



(a) Fashion-MNIST



(b) ShapeNet

Figure 9: Typical samples of hierarchical data on Fashion-MNIST (a) and ShapeNet (b). Images within a purple rectangular box are some instances of a leaf-level category. Categories within a green rectangular box belong to one common super-category. The super-categories within a red rectangular box share one common ancestor. On ShapeNet, categories within one purple rectangular box can be further divided into four child categories based on pose variations. Therefore, one hierarchy named Shape-C (Category) and another one named ShapeNet-P (Pose) are defined.

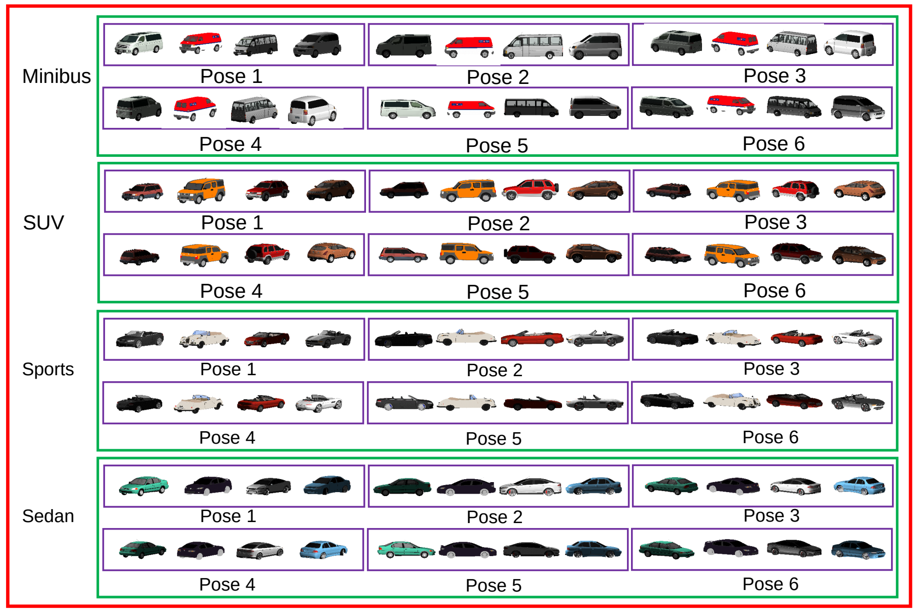
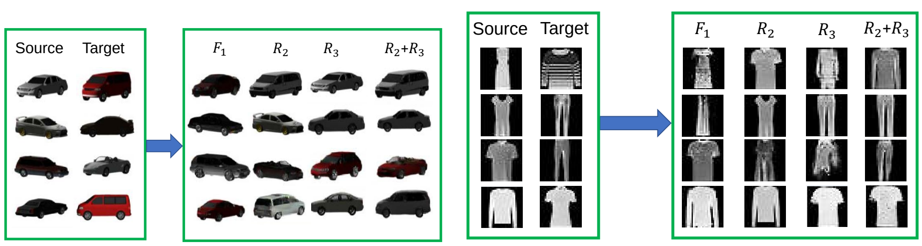
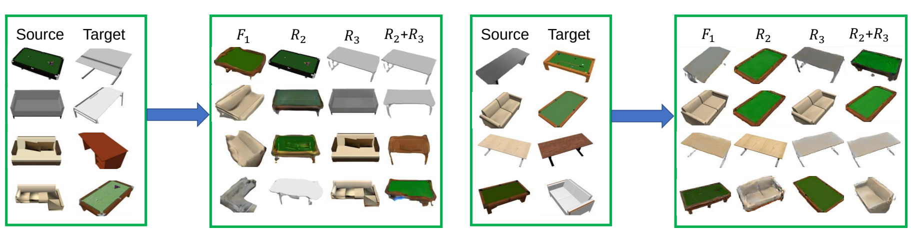


Figure 10: Typical samples of hierarchical data on CADCars. Images within a purple rectangular box are some instances of a leaf-level category. Categories within a green rectangular box belong to one common super-category. The super-categories within a red rectangular box share one common ancestor.



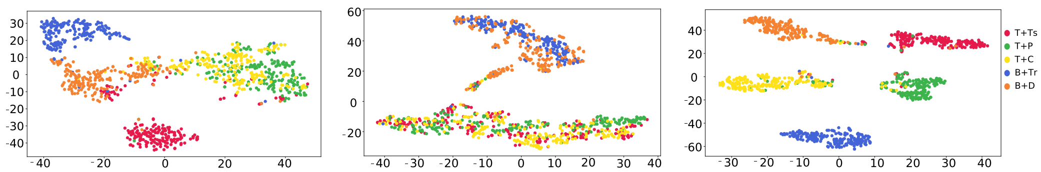
(a) CADCars (b) Fashion-MNIST

Figure 11: Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on CADCars(a) and Fashion-MNIST (b).



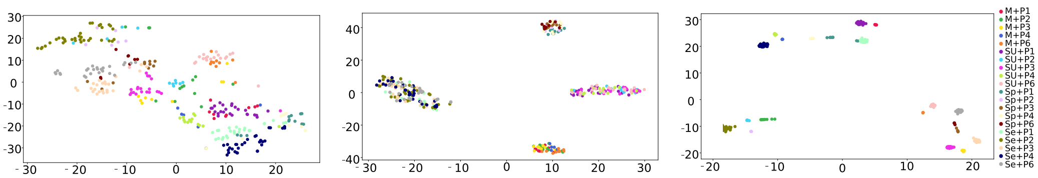
(a) ShapeNet-C (b) ShapeNet-P

Figure 12: Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on ShapeNet-C (a) and ShapeNet-P (b).



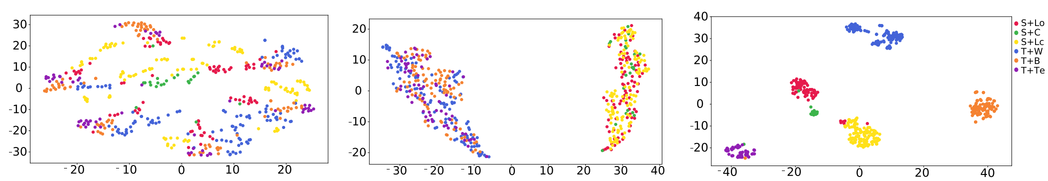
(a) Root (b) Level 2 (c) Level 3

Figure 13: 2D tSNE of disentangled $\mathbf{F}_l$ on test set of Fashion-MNIST for different levels. T and B mean Top and Bottom, Ts, P, C, Tr and D mean T-shirt, Polo shirt, Coat, Trousers and Dresses respectively.



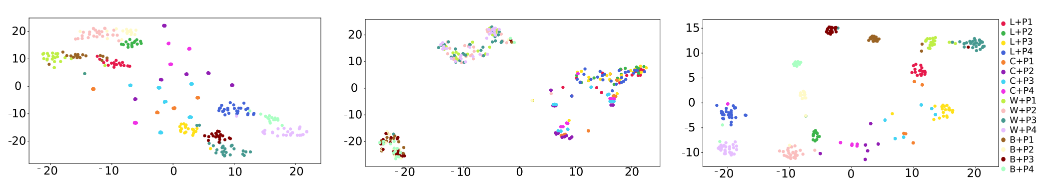
(a) Root (b) Level 2 (c) Level 3

Figure 14: 2D tSNE of disentangled $\mathbf{F}_l$ on test set of CADCars for different levels. M SU, Sp and Se mean Minibus, SUV, Sports and Sedan, P1$\sim$ P4 and P6 mean Pose 1, 2, 3, 4 and 6 respectively.



(a) Root (b) Level 2 (c) Level 3

Figure 15: 2D tSNE of disentangled $\mathbf{F}_l$ on test set of ShapeNet-C for different levels. S and T mean Sofa and Table, Lo, C, Lc, W, B and Te mean Loveseat, Club chair, L-couch, Work table, Billiards and Tennis-table respectively.



(a) Root (b) Level 2 (c) Level 3

Figure 16: 2D tSNE of disentangled $\mathbf{F}_l$ on test set of ShapeNet-P for different levels. L, C, W and B mean Loveseat, Club chair, Work table and Billiards respectively.