Pattern Recognition 140 (2023) 109539

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Hierarchical disentangling network for object representation learning

Shishi Qiao [a,b,c], Ruiping Wang [b,c,d], Shiguang Shan [b,c], Xilin Chen [b,c,*]

[a] College of Information Science and Engineering, Ocean University of China, QingDao 266100, China
[b] Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
[c] University of Chinese Academy of Sciences, Beijing 100049, China
[d] Beijing Academy of Artificial Intelligence, Beijing 100084, China

ARTICLE INFO

Article history:
Received 9 April 2022
Revised 4 January 2023
Accepted 19 March 2023
Available online 21 March 2023

Keywords:
Object understanding
Hierarchical learning
Representation disentanglement
Generative adversarial network
Network interpretability

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 natural hierarchical characteristics, i.e., any two objects can share some common primitives at a particular category level while possess unique traits at another. However, previous works usually operate in a flat manner (i.e., at a particular level) to disentangle the representations of objects. Even though they may obtain the primitives to constitute objects as the categories at that level, their results are obviously not efficient and complete. In this paper, we propose a Hierarchical Disentangling Network (HDN) to exploit the rich hierarchical characteristics among categories to divide the disentangling process in a coarse-to-fine manner (i.e., level-wise), such that each level only focuses on learning the specific representations and finally the common and unique representations at all levels jointly constitute the raw object. Specifically, HDN is designed based on an encoder-decoder architecture. To simultaneously ensure the level-wise disentanglement and interpretability of the encoded representations, a novel hierarchical Generative Adversarial Network (GAN) is introduced. Quantitative and qualitative evaluations on popular object datasets validate the effectiveness of our method.

1. Introduction

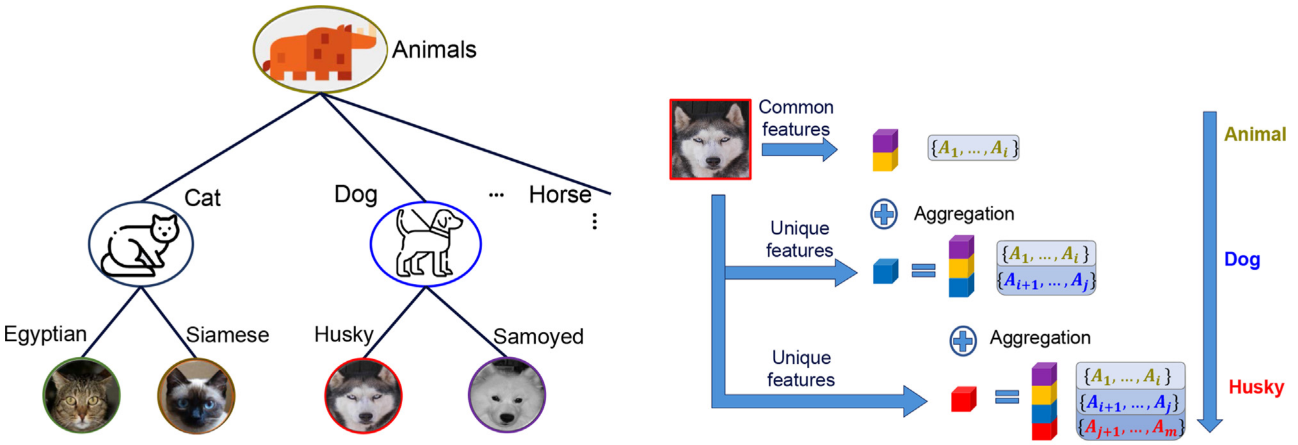
Representation learning is a basic and hot topic in machine learning and computer vision community, which has achieved significant progress in the recent years on different tasks such as recognition [1], detection [2–4] and generation [5], benefiting from the rapid development of representation learning by deep neural networks. Considering the strong capacity of deep neural networks, in this paper, we mainly focus on the deep representation learning framework.

Despite great success the deep representations have achieved, two important problems are still unresolved or less considered, i.e., the interpretability and the disentanglement of the learned representations. In the past decade, various works have been developed to reveal the black box of deep learning [6–11] and move us closer to the goal of disentangling the variations within data [12–18]. Even though they have brought great insights to us, they still have some limitations. For instance, Chen et al. [18], Xie et al. [19], Zhao et al. [20] 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 [16,17] try to obtain the domain-invariant and domain-specific knowledge, they can only handle two categories at a time, which is not that efficient. Besides, previous works mainly learn representation of objects in a flat manner, i.e., in a specific categorical level, which may not be flexible and complete. In this paper, we attempt to learn disentangled representations in a more natural and efficient manner.

Let us recall how humans understand an object. Generally speaking, an object is usually described as the combination of many semantic attributes, e.g., a *Husky* is a quadruped furry animal with bulged frontal bone and representative gray-white texture. Hundreds of thousands of objects in the world can be clustered and recognized by humans just because we can figure out the common and unique attributes of an object compared to the others. For example, one person who has never seen the Husky can recognize it as a *dog* in terms of its four legs, furry and bulged frontal bone features, while an animal expert may regard it as a

[*] Corresponding author at: Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China.
E-mail address: xlchen@ict.ac.cn (X. Chen).

https://doi.org/10.1016/j.patcog.2023.109539
0031-3203/© 2023 Published by Elsevier Ltd.



(a) Hierarchical structure

(b) Hierarchical features extraction

Fig. 1. Illustrations of (a) a hierarchical structure and (b) extracting the hierarchical features that constitute an object image. 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. The information encoded at each level is the semantic attributes A_i for describing the object in that granularity.

Husky by further observing its unique gray-white double coat. Both of them are right since categories have natural hierarchical structure and one can understand the object at any level.

As shown in Fig. 1(a), given four leaf-level categories, they are organized in a three-level hierarchical structure, considering the common and different semantics (e.g., visual attributes) they have. Each child category (e.g., the *Husky*) in the hierarchy is a special case of its parent category (i.e., the *dog*), since it inherits all attributes from its parent category and has extra variations that are not owned by its parent category. From another perspective, each parent category is the abstraction of all its child categories, considering it contains basic attributes that are present in all its child categories. Then we come back to the task of object representation learning. It aims to learn the representation encoding useful information that can be applied to other tasks (e.g. building classifiers and predictors) [21]. Taking the hierarchical nature of categories into account, if we only learn the representations of an object in a flat manner at a specific category-level as most previous works do, it will not be scalable and comprehensive for the machine to accomplish various tasks in the real world, such as the open world unseen object understanding, as shown in the experiments.

Our work aims to exploit such natural hierarchical characteristics among categories to divide the representation learning in a coarse-to-fine manner (i.e., level-wise), such that each level only focuses on learning that granularity of representation. For instance, given an object image in Fig. 1(b), it tangles the information of being an *animal*, a *dog* and a *Husky*. To achieve the objective of hierarchical disentangling and simultaneously interpreting the results so that humans can understand, we propose the Hierarchical Disentangling Network (HDN), which draws lessons from hierarchical classification and the recent proposed conditional generative adversarial nets (cGANs) [22]. Specifically, we first extract the features that only contain the information of being the *animal* from the image. By tracing from the root to leaf level, more and complementary information is extracted until we can recognize its belonging categories at all hierarchical levels. Given these disentangled visual primitives, they can be recombined from different objects as conditions and then visualized in the image space (e.g., features of a *Siamese* being the cat plus features of a *Husky* being different from the Samoyed should generate a Husky-like cat) via a conditional generative model. In the image space, hierarchical classification loss is adopted to constrain the uniqueness of each level, i.e., minimizing the cross entropy between the recombined features at different levels and corresponding semantic changes of the generated images.

By doing so, the disentangled representations of HDN are expected to find wide and promising applications. For example, one can change the semantics of a source object to those of a target at a specific category level while keeping information of other levels unchanged, e.g., the semantic controlled image-to-image translation in the experiments. Besides, it would help for the hierarchical image retrieval task using different levels of the disentangled representations, as shown in a case of attributes retrieval in our experiments. Apart from these, extensive experiments are conducted on several popular object datasets to validate the disentangling effectiveness of our method.

We summarize the main contributions of this paper in the following:

- We address the object representation learning problem from the generative perspective, i.e. what are the visual primitives constituting an object. Different from existing works, this manuscript focuses on a more natural and complete understanding of objects, i.e. dissect the object at different semantic levels inspired by the knowledge of taxonomy. By doing so, we obtain more general level-wise representations of objects which can be applied to several discriminative and generative downstream tasks.
- We propose a two-branch level-wise disentangle framework, which disentangles the constitutes of objects into basic commonality and hierarchical individuality parts. In the individuality branch, the novel semantic combination scheme is leveraged to ensure disentanglement of representations at different levels.
- We propose a hierarchical generative adversarial network to supervise the representation learning. The main difference from previous conditional GAN-based methods is that the semantic conditional inputs are consisted of multiple coarse-to-fine parts, each of which captures a local distribution of the whole data manifold. To this end, a hierarchical auxiliary classifier is elaborately designed accompanying with the discriminator.
- We conduct extensive experiments consider both generative and discriminative evaluations of the learned general representations at different semantic levels. Specifically, we conduct evaluations of object recognition including general, open-

world and cross domain scenarios, semantic retrieval and attributes/categories manipulation/translation in terms of both quantitative and qualitative metrics.

2. Related work

Our goal is to learn level-wise disentangled and interpretable representations for a specific object with deep networks, under the guidance of hierarchical prior. Therefore, our work is mainly related to disentangling deep representations, network interpretability and hierarchy-regularized learning.

Disentangling deep representations The goal of disentangling representation learning is to discover factors of variation within data [21]. Recent years have witnessed a substantial interest in such research area [23], including the works based on deep learning [12–18,24–26]. Rifai et al. [14] is probably the earliest to learn disentangled representations using deep networks for the task of emotion recognition. Reed et al. [12] is based on a higher-order Boltzmann machine and regards each variation factor of the manifold as its sub-manifold. Recently, Mathieu et al. [13], Chen et al. [18], Alharbi and Wonka [27], Deng et al. [28], Shen et al. [29], Mu et al. [30], Wadhwani and Awate [31] leverage the generative adversarial nets (GAN) to learn factors of variations to control the semantic of image synthesis. The cross-domain translation methods [16,17,32] learn the domain-specific representations to realize domain transfer.

However, these works ignore the natural and inherent hierarchical 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 addition, most existing works focus on dimension-wise disentanglement for independent factors such as pose, lighting, font width and so on, using simple datasets. As for complex real-world images, many factors can be correlated with each other and as a whole to represent a conceptual variation, where dimension-wise disentanglement has not been well studied. To address these issues, Tong et al. [25], Kaneko et al. [26] propose to learn the multivariant variables for modeling data variations using multi-dimensional vectors to represent difficult conceptual variations. To be specific, they progressively factorize such complicated variables into several mutually exclusive groups with narrowed variations, leveraging the hierarchical inclusion relationship. Each group at lower level focuses on a particular finer-grained conceptual variation.

Tong et al. [25], Kaneko et al. [26] possess similarities with ours on the disentangling manner, i.e., level-wise disentanglement learning. Nevertheless, they have substantial differences from ours. Specifically, Tong et al. [25] is a discriminative model to learn semantic, non-semantic and discriminative features capturing the fine-grained difference among categories for zero-shot learning task, and [26] is a generative model which aims to learn the factor-controlled generation progress like [18,27,28]. In other words, these two works focus on analyzing data variations from the perspective of the *whole data manifold* by dividing the complex variations at high level into more controllable and finer-grained ones at low levels for *specific tasks*, and thus the semantic of representation at high level contain that at low level. In contrast, ours is a generative model and *task-agnostic*, aiming at *single object* understanding in the human-like manner by dividing the constitution of objects into multiple complementary semantic parts, and thus the semantics of representation at high level are the subset of that at low level. Besides, the disentangled features of our method are more general and can serve for downstream tasks, as we validate in the experimental section.

Network interpretability Network interpretability aims to learn how the network works via visualizing it from the perspective that humans can understand. Related methods can be 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 [6–8,33,34], interpret the area in an image that is responsible for the network prediction [9,10,35–39], or manipulate the attributes of generated image by disentangling the latent space [29]. 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 [11,40,41]. However, these methods mainly focus on figuring out the running mechanism of networks while paying less attention to dissect variations among categories, which cannot ensure the models really understand their inputs.

Hierarchy-regularized learning Semantic hierarchies have been explored on object classification task for accelerating recognition [42,43], obtaining a sequence of predictions [44,45], making use of category relation graphs [46,47], and improving recognition performance through additional supervision [25,48–53]. While these discriminative classification works have achieved their expected goals, they usually lack interpretability. To address such issues, Xie et al. [19], Zhao et al. [20] 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. [54] uses an unsupervised generative framework to hierarchically disentangle the background, object shape and appearance from an image, and [26] attempts to capture the granularity-controlled conditions for image synthesis with decision tree latent controller. 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.

Our work lies in the intersection of above three research areas, and jointly exploits their advantages of level-wise disentangling variant and invariant factors within data in an efficient coarse-to-fine manner, as well as interprets them in the human-understandable image space via the generative learning framework.

3. Hierarchical representation learning

3.1. Problem formulation

Supposing that a category hierarchy is given in the form shown in Fig. 1(a), we use $l = 1, \ldots, L$ 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 of 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 $\mathbf{F}_l$ at the l-th level.

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

$$\mathbf{O} = \underbrace{\underbrace{\underbrace{\{\mathbf{A}_1, \ldots, \mathbf{A}_i\}}_{level=1} \cup \{\mathbf{A}_{i+1}, \ldots, \mathbf{A}_j\}}_{level=2 \quad \ldots} \cup \{\mathbf{A}_{j+1}, \ldots, \mathbf{A}_m\}}_{level=L} \cup \Delta \tag{1}$$

where Δ represents currently undefined attributes existing on $\mathbf{O}$. As we have discussed, humans classify $\mathbf{O}$ at a particular category level according to a subset of the whole attribute set in Eq. (1). Take the object in Fig. 1(b) for example, it can be regarded as an *animal* since it contains the attribute subset $\{\mathbf{A}_1, \ldots, \mathbf{A}_i\}$, and be classified to a *dog* in terms of the attribute subset $\{\mathbf{A}_1, \ldots, \mathbf{A}_i, \mathbf{A}_{i+1}, \ldots, \mathbf{A}_j\}$ present in it. Therefore, the disentangled feature $\mathbf{F}_l$ for our objectives in Fig. 1(b) should encode the information of the attribute subset formulated in Eq. (1). Moreover,

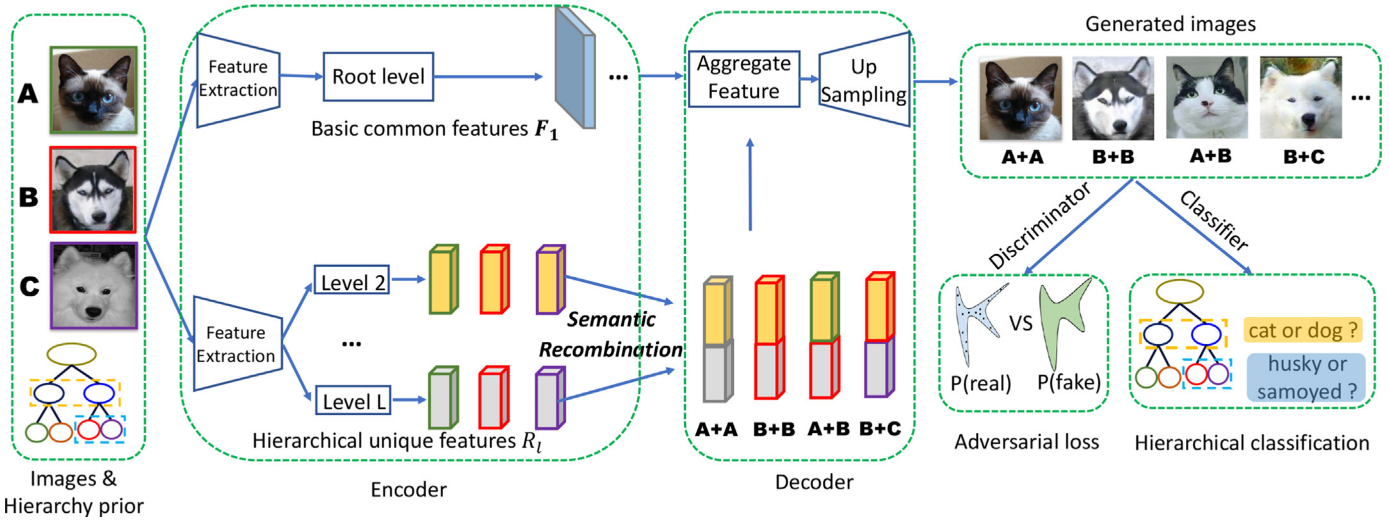


Fig. 2. An illustration of the framework of our method. Assume images A, B and C belong to a three-level hierarchy in Fig. 1 (not limited by three, one can add more levels), the root-level common feature of their being the root category and the unique features at non-root levels that can further distinguish them as finer-grained categories are extracted by the upper and bottom convolutional branches, respectively. To ensure semantically disentangling and human friendly interpretability of the unique features, different levels of them from the same or different objects are randomly recombined, and then reconstructed via feature aggregation (Adaptive Instance Normalization (AdaIN)) and conditional generation in the image space, where adversarial loss and hierarchical classification loss are elaborately designed.

due to the hierarchical correlations (i.e., the inherited relationship) among categories at different hierarchical levels, obviously the subset $\{\mathbf{A}_1, \ldots, \mathbf{A}_i, \mathbf{A}_{i+1}, \ldots, \mathbf{A}_j\}$ includes $\{\mathbf{A}_1, \ldots, \mathbf{A}_i\}$, naturally leading to the disentangled $\mathbf{F}_{l-1}$ being the proper subset of $\mathbf{F}_l$. In other words, at non-root levels, only the unique feature compared to its parent needs to be learned. With such hierarchical complementary characteristics, the unique features at different levels should reflect different aspects of the object appearance. If we randomly recombine each level of F_l from the same or different objects and reconstruct them in the image space, the appearance of generations would be changed by the semantic information encoded in $\mathbf{F}_l$, e.g., a generated image combines the shape of a cat and the texture of a Husky.

Taking these into consideration, we design the Hierarchical Disentangling Network (HDN) based on the autoencoder architecture in Fig. 2. The encoder E dissects the hierarchical representations given a semantic hierarchical prior. Different levels of unique features from the same (or different) objects are recombined and aggregated with the root-level feature as the conditions for the decoder G to reconstruct images. By doing so, the semantics of disentangled features at different levels can be visualized in the image space. During training, to ensure the consistency between the appearance changes of generations and the semantics encoded in recombined features from corresponding levels, the discriminator D and the hierarchical classifiers H (they share the frontal backbone architecture except the output layers) are designed.

3.2. Hierarchical representation extraction

In this subsection, we introduce the detailed extraction process of $\mathbf{F}_l$. 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 such information, 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. Therefore, for implementation of hierarchical disentanglement, only the common feature $\mathbf{F}_1$ at the root level and the unique ones $\{\mathbf{R}_l\}_{l=2}^{L}$ at deeper levels are extracted in this paper.

To ensure the semantics of these features and interpret them to humans, 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 basic content of the object, while those of $\{\mathbf{R}_l\}_{l=2}^{L}$ are unique for different levels which play the role of the variant styles of the object. Therefore, $\mathbf{F}_1$ and $\{\mathbf{R}_l\}_{l=2}^{L}$ are processed in the upper and bottom branches respectively to make them play different roles during the reconstruction, as shown in Fig. 2.

3.3. Constraints for the learning process

The basic constraints on hierarchical disentanglement are making features at different levels perform their own duties. For an object $\mathbf{O}$, the encoded $\mathbf{F}_1$ and $\{\mathbf{R}_l\}_{l=2}^{L}$ should be 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 for describing its being the root category. Progressively involving $\mathbf{R}_l$, one can distinguish it from other categories at the l-th level.

Apart from the semantic disentanglement, visualization of features in the image space such that we can figure out what has been encoded would be more human friendly. To kill two birds with one stone, we turn to the popular conditional generative adversarial nets (cGANs) [22] which can control generated images based on different semantic condition inputs. To be more specific, our HDN leverages the disentangled features $\mathbf{F}_1$ and $\{\mathbf{R}_l\}_{l=2}^{L}$ to control the variations of reconstructed images at different category levels. In return, by conducting loss functions on generated images, the semantics of the input conditions (i.e., $\mathbf{F}_1$ and $\{\mathbf{R}_l\}_{l=2}^{L}$) can be disentangled.

To ensure $\mathbf{F}_1$, $\{\mathbf{R}_l\}_{l=2}^{L}$ are well disentangled and complementary, we assume that the disentangled result at any level of one object can be exchanged with the one at the same level of another object, and such exchanged features at that level would be reflected on the appearance change of the reconstructed image (e.g. the leaf-level feature of a Siamese is replaced by that of a Husky may generate a Husky-like cat). Based on this idea, we propose to randomly recombine features at each level from two different objects and control the generated images through these combined features, as shown in Fig. 2. Specifically, given $\mathbf{F}_1^A$, $\{\mathbf{R}_l^A\}_{l=2}^{L}$ and $\mathbf{F}_1^B$, $\{\mathbf{R}_l^B\}_{l=2}^{L}$ disentangled from objects $\mathbf{O}_A$ and $\mathbf{O}_B$, we obtain the newly combined features $\mathbf{F}_1^{'}$ and $\{\mathbf{R}_l^{'}\}_{l=2}^{L}$. For each level, $\mathbf{R}_l^{'}$ ($\mathbf{F}_1^{'}$ if $l = 1$) comes from either $\mathbf{O}_A$ or $\mathbf{O}_B$. The newly combined features are aggregated together as the inputs for the decoder G to generate new object images $\mathbf{I}^g$. Such images should satisfy the following loss functions:

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

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

  where J_{cls} is a cross-entropy loss among local brother categories at each level that 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 $G(\mathbf{F}_1^{'}, \{\mathbf{R}_l^{'}\}_{l=2}^{L})$. $H(\mathbf{I}^g)_l^c$ is probabilistic prediction on the c-th local category, and y_l^c is the ground truth local label of the generated object at the l-th level.

  Please note that we only focus on the *local* brother categories (i.e. the ones belonging to the same parent category) instead of all categories at that level. It makes the disentanglement more flexible. On one hand, the classification at 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 classifiers 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 [55] loss is adopted in light of its stable training process, defined as:

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

- **Image reconstruction loss**. As for $\mathbf{F}_1^{'}$ and $\{\mathbf{R}_l^{'}\}_{l=2}^{L}$ from one same object, we should be able to reconstruct it as close to the input as possible.

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

  where $p'(G)$ is the distribution of generations taking $\mathbf{F}_1^{'}, \{\mathbf{R}_l^{'}\}_{l=2}^{L}$ 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.

$$J_{recon}^{\mathbf{F,R}} = \mathbb{E}_{(\mathbf{F}_1^{'}, \{\mathbf{R}_l^{'}\}_{l=2}^{L}) \sim p(E)} [||E(G(\mathbf{F}_1^{'}, \{\mathbf{R}_l^{'}\}_{l=2}^{L})) - (\mathbf{F}_1^{'}, \{\mathbf{R}_l^{'}\}_{l=2}^{L})||_1] \tag{6}$$

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

Now we combine the four loss functions defined in Eqs. (3)–(6) into one comprehensive loss function for supervising the training of the proposed method:

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

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

As for the update of the discriminatorD and hierarchical classifiers H, they are optimized by the following loss:

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

3.4. Implementation details

Our HDN is implemented on Pytorch platform.[1] Design of the backbone follows recent proposed image generation [56] and image-to-image translation works [17]. Images are resized to 128 × 128 resolution for all datasets except Fashion-MNIST which is resized to 28 × 28.

For aggregation of the common and unique features, i.e., $\mathbf{F}_1^{'}$ and $\{\mathbf{R}_l^{'}\}_{l=2}^{L}$, we equip the residual blocks with the Adaptive Instance Normalization (AdaIN) [57], the parameters of which are dynamically generated by a multi-layer perception (MLP) from the disentangled unique features. To be specific, the recombined $\{\mathbf{R}_l^{'}\}_{l=2}^{L}$ are concatenated first, and then aggregated with $\mathbf{F}_1^{'}$ by:

$$AdaIN(\mathbf{F}_1^{'}, \gamma, \lambda) = \gamma \left(\frac{\mathbf{F}_1^{'} - \mu(\mathbf{F}_1^{'})}{\sigma(\mathbf{F}_1^{'})} \right) + \lambda \tag{9}$$

where μ and σ are channel-wise mean and standard deviation, γ and λ are generated by the MLP from the concatenated unique features. No normalization is used in the bottom encoder branch. We adopt ReLU activation 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 resolution) are used to ensure both realistic details and global structure. The last layer of the decoder is equipped with a *tanh* activation to normalize the values of generated images to the range of $[-1, 1]$. More network details are given in the supplementary material.

During training, 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 300 K iterations and half decay the learning rate every 100 K iterations. We set batch size to 16. The loss weights α and β in Eq. (7) are set as 10 and 1 respectively, following the settings of Huang et al. [17]. Random mirroring is applied during training.

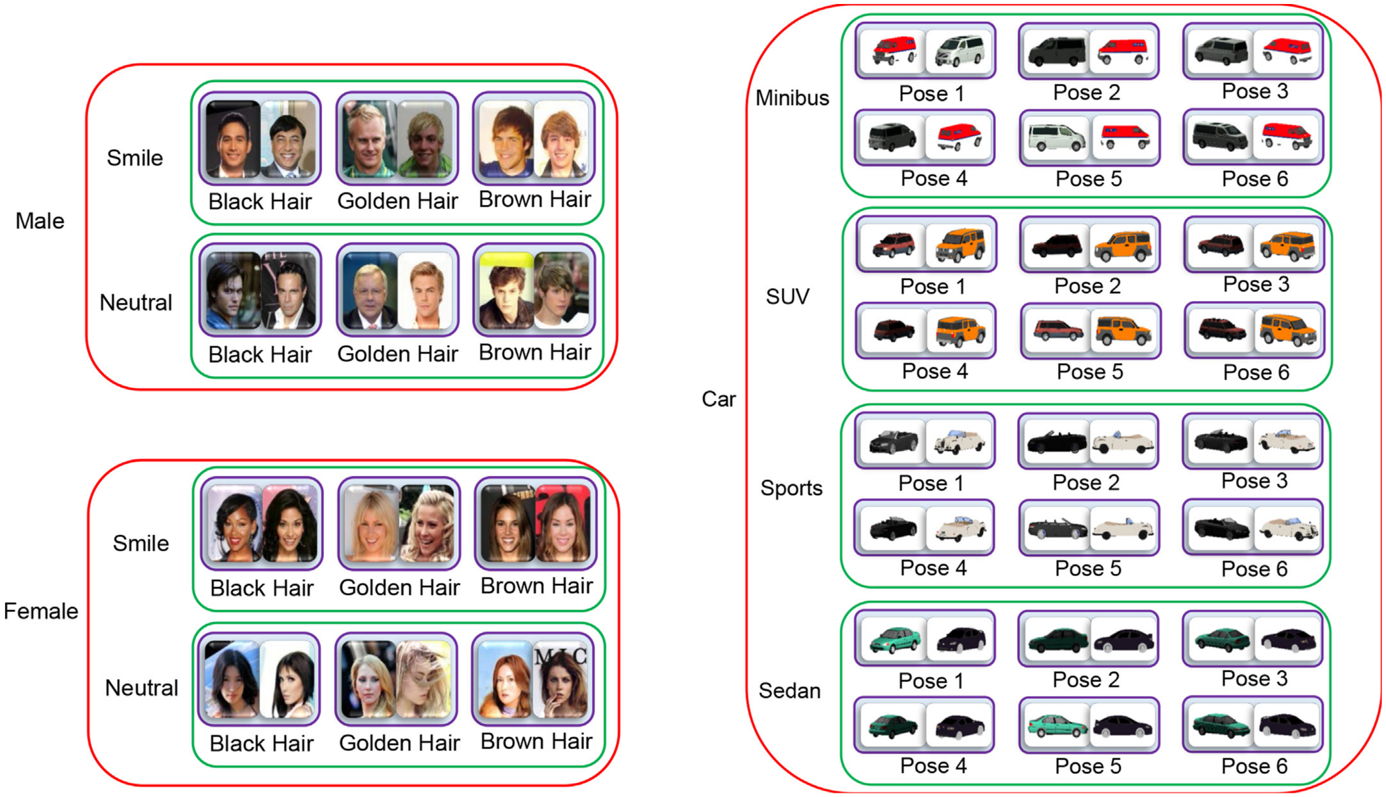
4. Model disentangling analysis

Datasets We conduct experiments on hierarchical annotated data from four datasets, typical examples are shown in Figs. 3 and 4.[2] The first is CelebA dataset [58]. It provides more than 200 K face images and 40 attribute annotations. Following the official train/test splits, we define a four-level hierarchical 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 at the next two levels. With such ground-truth hierarchical annotations, we can validate our method more easily.

The second dataset named Fashion-MNIST [59] is proposed as a direct drop-in replacement of 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

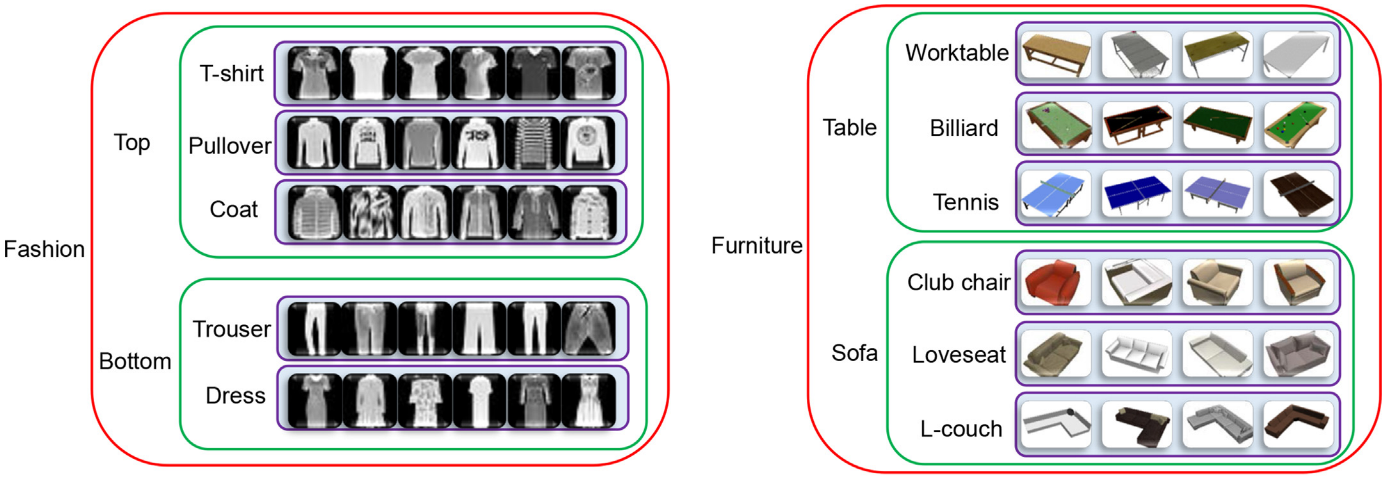
[1] The source codes will be released to the public.

[2] It is noted that the focus of this paper is to interpret the hierarchical structure within data. Therefore, we heuristically construct hierarchical structures based on two principles. First, it meets with the human perception, i.e., similar images should belong to one cluster. Second, it can be easily evaluated in the experiments based on the hierarchical descriptions in Eq. (1), i.e., we can clearly observe and describe the differences between levels in the hierarchy for an object image. One can also automatically obtain reasonable hierarchical annotations using machine learning technologies such as unsupervised clustering as Goo et al. [52] does.



(a) CelebA (b) CADCars

Fig. 3. Typical samples of hierarchical data on CelebA (a) and CADCars (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. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Fashion-MNIST (b) ShapeNet

Fig. 4. 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 (Furniture → Common Categories → Fine-grained Categories) and another one named ShapeNet-P (Furniture → Fine-grained Categories → Pose Variations) are defined. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

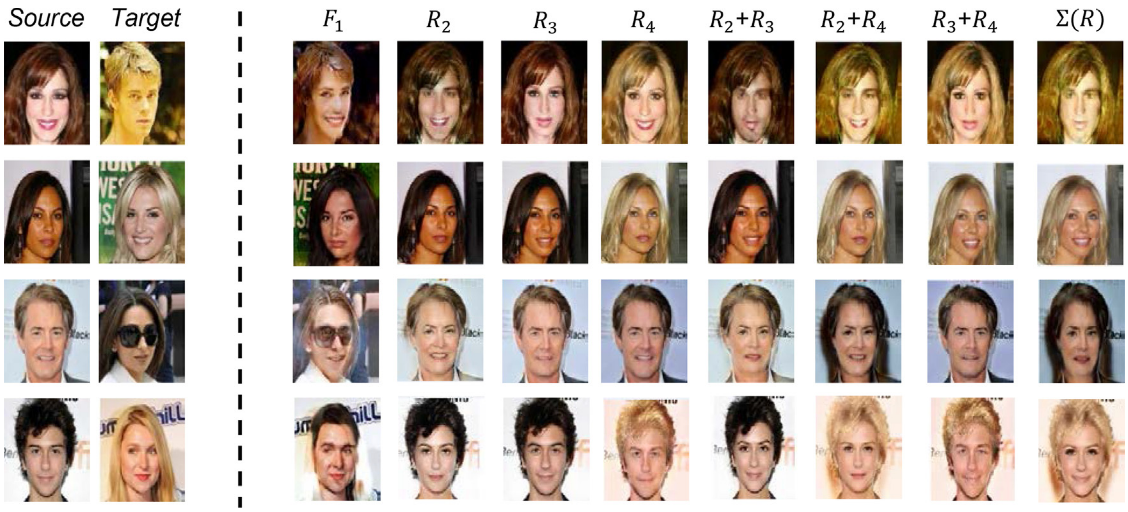


Fig. 5. Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on CelebA. Different columns denote results of using $\mathbf{F}_1$, $\{\mathbf{R}_l\}_{l=2}^{L}$ or their combinations disentangled from the target images to replace the corresponding levels of the sources (+ and $\sum$ mean which levels participate in the exchange, rather than the numerical summation). Ground truths of $\mathbf{R}_2, \mathbf{R}_3, \mathbf{R}_4$ are gender, smile and hair color variations, respectively.

hierarchical structure, we cluster T-shirt, coat, pullover as one super category, and trouser, dress as another super one to construct a three-level hierarchical structure (root is fashion) according to their appearance similarity.

The other two datasets are 3D data, i.e., CADCars [60] and ShapeNet [61]. CADCars contains 183 3D Car models, and ShapeNet is constitutive of 51,300 3D models covering 55 common and 205 finer-grained categories. Using their provided tools, we generated 24 2D images with 6 poses 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 hierarchical structure. On ShapeNet, 12 2D images with pose variation are obtained for each 3D model. One three-level category-pose hierarchical structure named ShapeNet-P (i.e., Furniture → Fine-grained Categories → Pose Variations) and another three-level hierarchical structure named ShapeNet-C (i.e., Furniture → Common Categories → Fine-grained Categories) are defined. The ratio of train/test split is 4:1 by random divisions.

4.1. Disentangling results

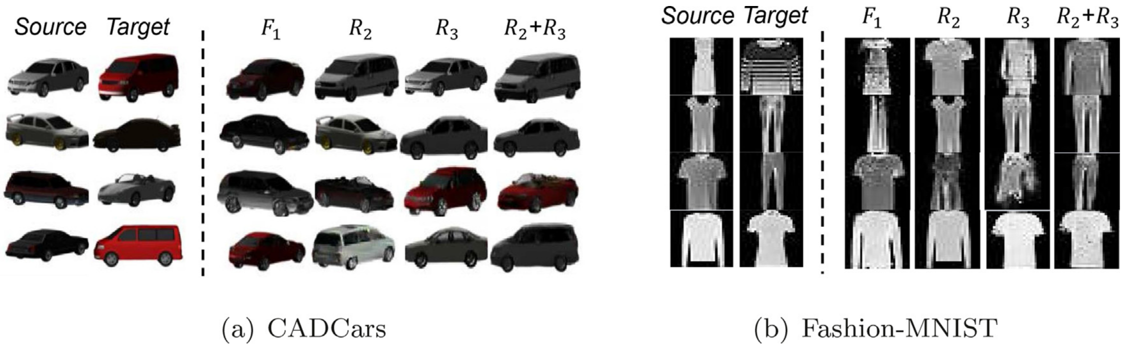
As introduced in Section 3.3, the recombined features at each level may come from different input objects, and the appearance of generated image should reflect the semantic information encoded in each level. Therefore, in this part, we first replace one or multiple levels of disentangled features of a source image with those of a target image, and then observe the visual changes of generated image to validate the semantic consistence with pre-defined hierarchies.

Figs. 5–7 show such semantic translation results. It is observed that different level of features perform their own duties, i.e., they carry just enough information to control the variations at that level (e.g., gender, smile and hair color from the second to leaf levels on CelebA we specially predefined), but would not involve more information that should belong to other levels. For instance, in Fig. 5 as we replace features of an image at any one, two or all levels with those of another image, the semantics would be changed correspondingly. Apart from expected disentanglement of unique features $\{\mathbf{R}_l\}_{l=2}^{L}$, 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). In Figs. 6 and 7, the disentanglement becomes tougher to some extent, as the variations at some levels are categorical such as the semantics of being a SUV at $\mathbf{R}_2$ level on CADCars, the difference between the trouser and dress at $\mathbf{R}_3$ level on Fashion-MNIST). Nevertheless, on CADCars and ShapeNet-P, our HDN can still accurately capture such semantics at each level, i.e., categorical difference at $\mathbf{R}_2$ level and pose variation at $\mathbf{R}_3$ level. On Fashion-MNIST and ShapeNet-C, the semantic uniquenesses at the second and third levels are coarse and fine-grained categorical differences respectively (e.g., table vs. sofa, and billiards vs. worktable). We can find that from $\mathbf{R}_2$ to $\mathbf{R}_2 + \mathbf{R}_3$, the appearances of source images are changed almost consistent with the hierarchy structure.

To give a more intuitive feeling about the hierarchical uniqueness of such level-wise features, we investigate the discriminabilites of them on CelebA via the popular tSNE tool [62]. As shown in Fig. 8, with only the common feature $\mathbf{F}_1$, samples are mixed together. When progressively aggregated with features at deeper levels $\mathbf{R}_l$, samples are better separated and almost consistent with the hierarchical structure, which further verifies our method has successfully disentangled the hierarchical semantics.

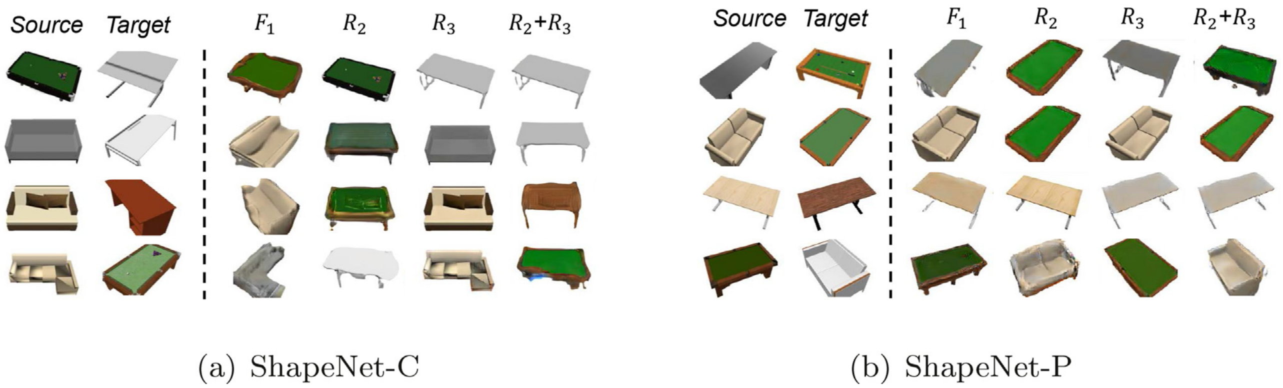
Apart from the direct level-wise feature exchange, we also show that one can transform the source image smoothly by linear interpolation (with 5 equally spaced interpolation coefficients from 0.1 to 0.9) of disentangled features between the source and target. Such examples are shown in Fig. 9. We can see that the genders, expressions, hair colors and their combinations on the source images (first columns in each case) can be changed smoothly towards those on the targets (last columns of each case). Learning a smooth feature space with continuous variations is a significant issue for representation learning, which can ensure the generalization ability for unseen similar objects. We have made a further investigation of such task in Section 5.2.

Finally, a quantitative evaluation of these results is conducted. Specifically, we use the learned hierarchical classifier H to evaluate whether hierarchical semantics are correctly disentangled, recombined and finally decoded into the generated images shown in Figs. 5–7. To ensure H is reliable, the accuracy of classifications at each level on real test images is given as a reference. Table 1 gives the evaluation results. Firstly, it can be seen that the semantics of generated images by changing different levels are recognized



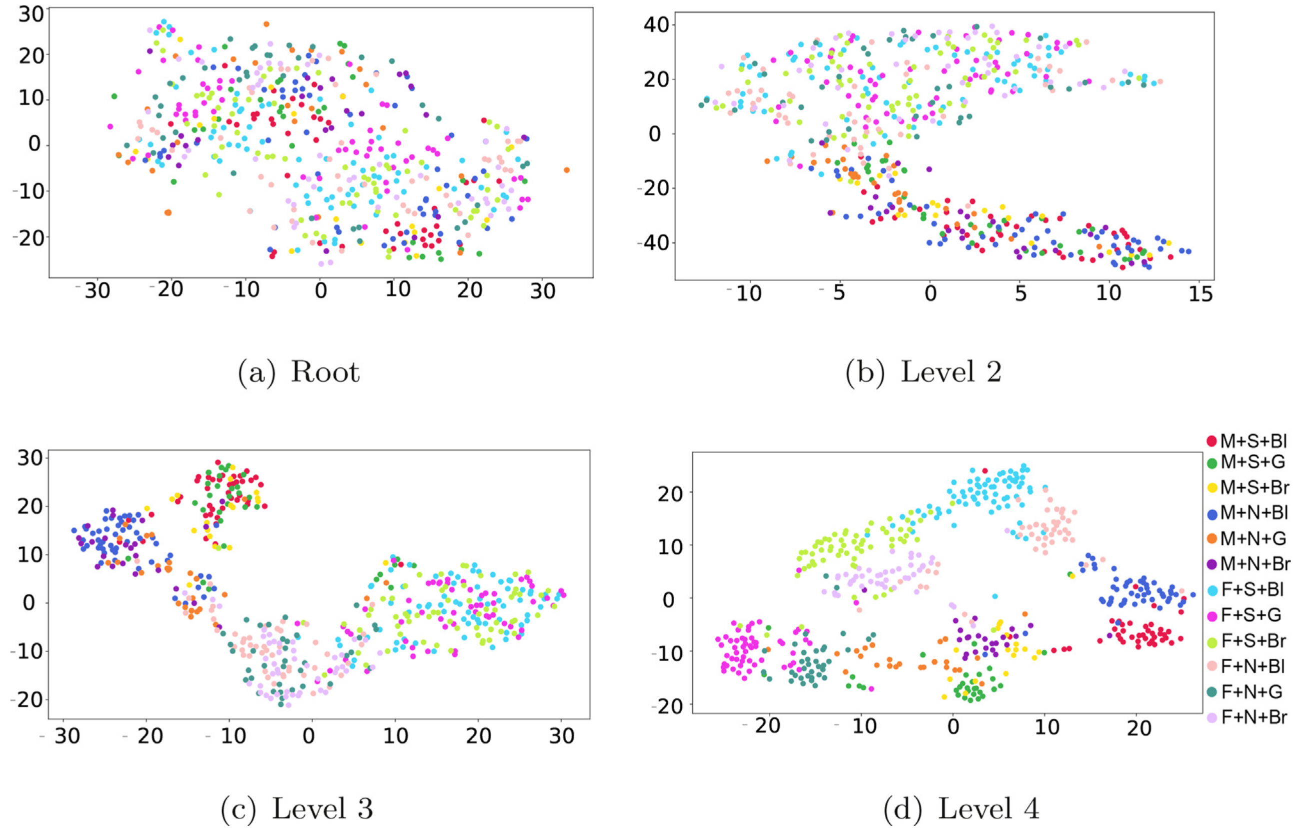
(a) CADCars (b) Fashion-MNIST

Fig. 6. Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on CADCars(a) and Fashion-MNIST (b). On these hierarchical data, only the leaf-level $\mathbf{R}_3$ of CADCars has describable ground truth (i.e., pose variation), other levels are complex categorical variation (e.g., the semantics of being a SUV at $\mathbf{R}_2$ level on CADCars, the difference between the trouser and dress at $\mathbf{R}_3$ level on Fashion-MNIST).



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

Fig. 7. Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on ShapeNet-C (a) and ShapeNet-P (b). On these hierarchical data, only the leaf-level $\mathbf{R}_3$ of ShapeNet-P has describable ground truth (i.e., pose variation), other levels are complex categorical variation (e.g., additional semantics of being a billiard table compared with being only a table).



(a) Root (b) Level 2

(c) Level 3 (d) Level 4

Fig. 8. 2D tSNE of disentangled $\mathbf{F}_l$ on test set of CelebA at different levels. For easy understanding, M and F mean male and female, S and N mean Smile and Neural, and Bl, G and Br mean Black, Golden and Brown hair, respectively.

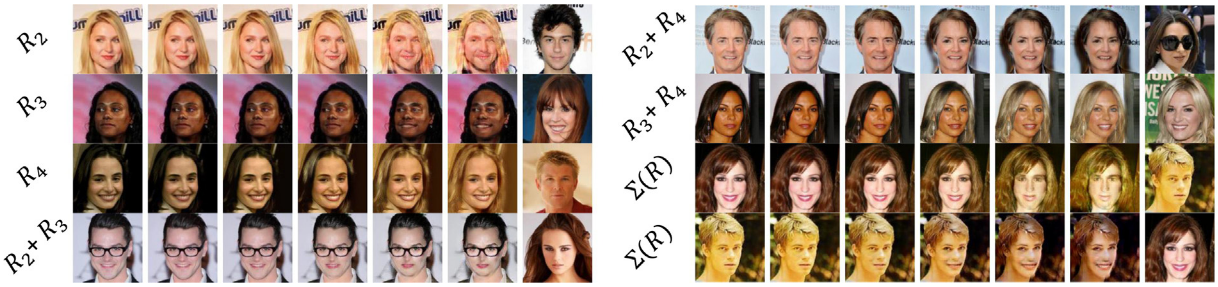


Fig. 9. Interpolations of disentangled $\mathbf{R}_l$ between the source (first columns) and target (last columns) images (other levels unchanged on the source images). Ground truths of $\mathbf{R}_2, \mathbf{R}_3, \mathbf{R}_4$ are gender, smile and hair color variations, respectively.

Table 1
Accuracy of hierarchical classifications for real images on test set and generated (Gen.) images using randomly recombined hierarchical features. Lv2 denotes the second level. CADCars-R denotes the reversed level order of Lv2 and Lv3 compared with CADCars.

	CelebA		Fashion-MNIST		CADCars		CADCars-R		ShapeNet-C		ShapeNet-P	
Level	Test	Gen.	Test	Gen.	Test	Gen.	Test	Gen.	Test	Gen.	Test	Gen.
Lv2	0.9570	0.9387	0.9629	0.9779	0.9781	0.9792	0.9798	0.9956	0.9941	0.9941	0.9323	0.8863
Lv3	0.9232	0.9103	0.9336	0.8464	0.9798	0.9670	0.9798	0.9219	0.9844	0.8865	0.9190	0.8413
Lv4	0.8932	0.8799	–	–	–	–	–	–	–	–	–	–

correctly. Secondly, the deeper of the level, the more difficult of the semantic change in general, since the criteria for distinguishing one category from others in the deeper level would become more and more fine-grained. Finally, it becomes difficult to transfer the unique features and to generate distinguishable images when that information is difficult to be described and disentangled at the leaf-level on Fashion-MNIST and ShapeNet-C (e.g., what it would look like by transferring the semantic difference between a *billiard* and a *tennis* to an *L-couch*), leading to poor classification accuracy at those levels.

4.2. Quality comparison of generated images

In this subsection, we evaluate the quality of generated images controlled by recombined hierarchical features on CelebA. Since the disentangling paradigm of our method is similar to the image-to-image translation task, we further compare one of such kinds of cGAN-based works, i.e. StarGAN [63] which has been a popular framework for the multi-attribute translation task. Besides, we also compare a specific face attribute disentanglement work named ELEGANT [32][3] We trained ELEGANT-2 for disentangling gender and smile, and ELEGANT-5 for all the 5 attributes we used in our HDN. We follow the hyper-parameters settings on CelebA in their publicly released codes. We use the Inception Score (IS) [64] and Frchet Inception Distance (FID) [65] to measure fidelity of images, and leverage the Learned Perceptual Image Patch Similarity (LPIPS) [66] to measure the diversity of generated visual modes to detect mode collapse.

In Table 2, it is observed HDN achieves comparable and even slightly better image fidelity compared with the state-of-the-art translation method StarGAN, which demonstrates that HDN can not only extract primitives of objects for discriminative tasks but also be applied to such graphical applications. Besides, we find that IS and LPIPS are sensitive to artifacts while FID is more stable, which can be validated by the qualitative results in the following.

Figure 10 compares the qualitative results. Our method performs comparable with StarGAN and better than ELEGANT. ELEGANT is designed for disentangling face attributes, the results of which for few attributes (no more than two as suggested by the authors) look good but would become much poor when multiple factors need to be dealt with, while ours can simultaneously handle multiple factors at different levels.

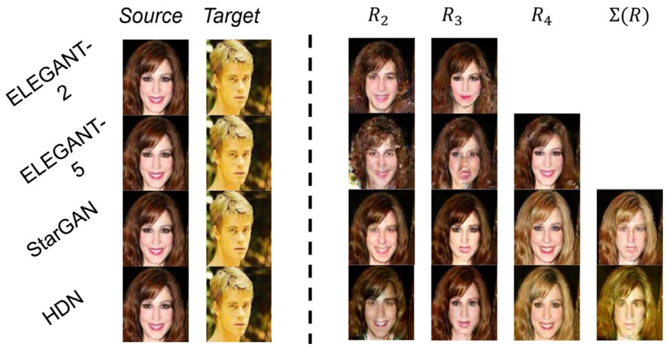


Fig. 10. Semantic translation results of compared cGANs and HDN on CelebA. $\mathbf{R}_2, \mathbf{R}_3, \mathbf{R}_4$ are gender, smile and hair color, respectively. StarGAN only needs binary attribute vectors as conditions to generate images, and the target image in its row is not used. ELEGANT-2 is trained with gender and smile attributes. When testing, the ELEGANT models can only change one attribute guided by the target each time.

4.3. Ablation study

In this subsection, we firstly make a justification of several choices made in our method, including the usage of *local* brother categories for classification learning, and different loss terms in Eq. (7) on the CelebA dataset. Specifically, we replace the local classification loss (i.e., multiple Softmax cross-entropy loss items are respectively computed on the categories which belong to one same parent category) with the global one (i.e., only one Softmax cross-entropy loss is computed on all categories) at each level to verify the effectiveness of local discriminability for disentangling level-wise unique features. For each loss term, we simply drop it and keep others unchanged during training. Furthermore, we investigate the relative loss weights of the hierarchical classification loss term that ensures the discriminability of learned features w.r.t the adversarial loss term that ensures the synthesis image quality. Specifically, we fix the weight of J_{GAN} as

[3] This method is good at disentangling two attributes in a model and changing one attribute with another fixed given a reference image each time. The performance would become unstable for more than two attributes, about which we have discussed with its authors.

Table 2
Comparisons of image quality of baselines, state-of-the-arts and the full HDN. IS and FID measure the fidelity, and LPIPS measures the diversity of images. For IS and LPIPS, higher is better. For FID, lower is better. w/o GAN, w/o cls, w/o fea, w/o img, and global denote HDN trained without the adversarial, the hierarchical classification, the feature reconstruction and image reconstruction loss terms, and with global classification loss, respectively. Real means the result of real images on test set.

	w/o GAN	w/o cls	w/o fea	w/o img	global	StarGAN	ELEGANT-2	ELEGANT-5	HDN-full	Real
IS	2.61	2.87	2.75	2.42	3.34	2.59	2.84	3.63	2.70	2.87
FID	86.24	14.37	20.70	28.87	77.35	20.19	25.78	51.6	20.07	0
LPIPS	0.439	0.412	0.411	0.408	0.430	0.409	0.404	0.499	0.408	0.416

Table 3
Accuracy of hierarchical classifications for real and generated images of baselines and our full method at different semantic levels on CelebA.

Level	w/o GAN	w/o cls	w/o fea	w/o img	global	HDN-full	Real
Lv2	0.9607	0.7385	0.9600	0.9434	0.8246	0.9387	0.9570
Lv3	0.9271	0.7224	0.9020	0.9068	0.7791	0.9103	0.9232
Lv4	0.8893	0.7450	0.8661	0.8662	0.8587	0.8799	0.8932

Table 4
Accuracy of hierarchical classifications at different levels for generated images in different relative weight settings of the classification loss w.r.t the adversarial loss on CelebA.

Level	0	0.001	0.01	0.1	1.0	10.0	100.0
Lv2	0.7385	0.8406	0.9309	0.9697	0.9387	0.9643	0.9614
Lv3	0.7224	0.7163	0.8187	0.8873	0.9103	0.9151	0.9203
Lv4	0.7450	0.7990	0.8287	0.8438	0.8799	0.8638	0.8675

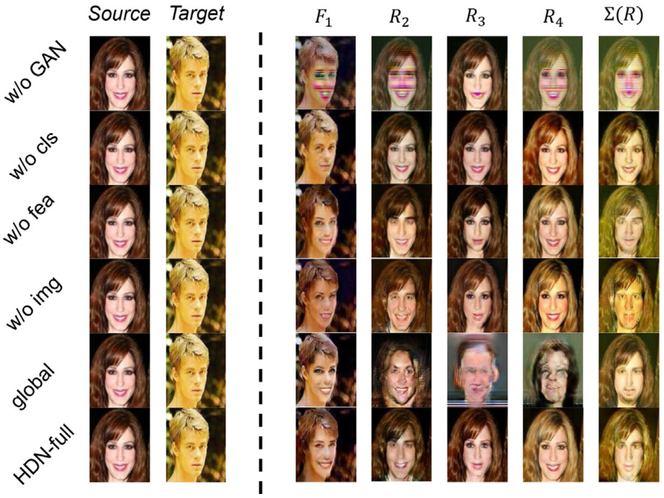


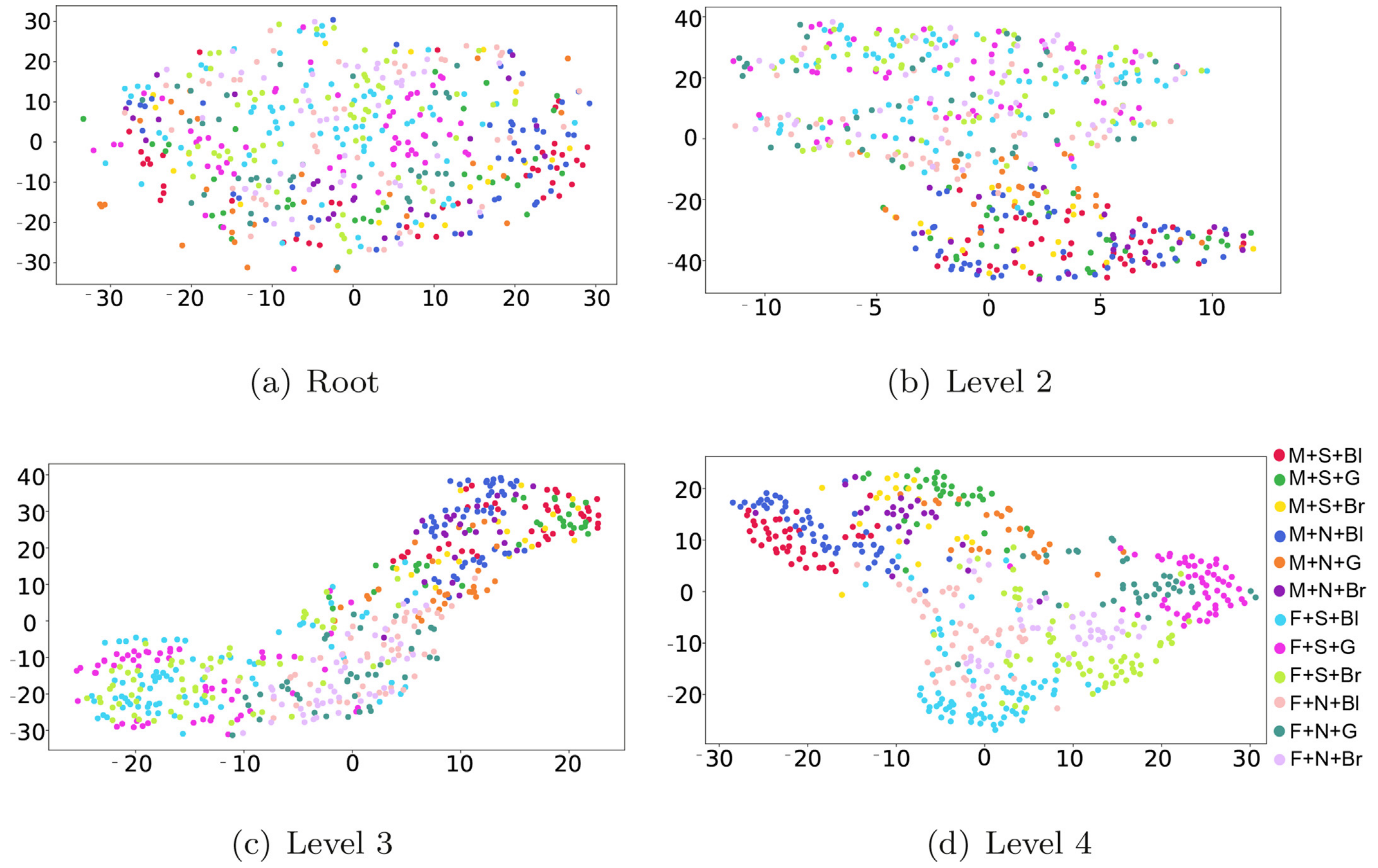
Fig. 11. Semantic translation results of baselines and HDN-full on CelebA. Different columns denote results of using $\mathbf{F}_1$, $\{\mathbf{R}_l\}_{l=2}^{L}$ or their combinations disentangled from the target images to replace the corresponding levels of the sources. Ground truths of $\mathbf{R}_2, \mathbf{R}_3, \mathbf{R}_4$ are gender, smile and hair color variations, respectively.

1.0 for reference, and change the weight of J_{cls} in the range of $\{0.001, 0.01, 0.1, 1.0, 10.0, 100.0\}$.

To evaluate the results, we firstly compare these baselines in terms of the classification performance in Tables 3 and 4, and the visual quality in Table 2, Figs. 11 and 13 for the generated images controlled by disentangled features. From Table 3, we can see that HDN-full overall performs better. Replacing the local classification loss with the global one at non-root levels would heavily do harm to the goal of hierarchical disentanglement, as the global one takes all categories at that level into consideration which needs the information at both parent and current levels, while we aim to separate such information (i.e., the unique feature at current level compared to its parent), leading to conflicting objectives. Such conflict leads to poor generation quality shown in Fig. 11. Without the local classification, only changing features of one level results in ambiguous generations (the fifth row “global” in Fig. 11), which can also be reflected from the quantitative evaluation of image quality in Table 2 (high FID). As for the reconstruction losses, they mainly stabilize the adversarial training. Without them, the quality of generated images would decrease to some extent. Besides, the feature reconstruction loss can boost the disentangling degree of features. As the 2D tSNE results in Fig. 12 demonstrates, without such loss, the intra-class compactness and inter-class discriminability of samples in the embedding space become poor.

As for the other two core losses, without the adversarial loss (i.e., w/o GAN), the quality of generated images in Table 2 is quite poor (high FID), even though the hierarchical classification accuracy in Table 3 is very high. This means that the decoder generates fake images with artifacts to cheat the classifier as shown in Fig. 11, demonstrating the necessary of adversarial loss to ensure the synthesis task. By contrast, without the classification loss (i.e., w/o cls), the decoder may generate images with high quality (very low FID) in Table 2, but without desired target semantics (poor classification accuracy), as shown in Table 3 and Fig. 11. Lastly, changing the relative loss weights of the classification loss w.r.t the adversarial loss from small to large values, the overall classification accuracy in Table 4 will be improved at the price of generated image quality decreasing in Fig. 13. As an empirical conclusion, setting the weight of the classification loss term as 1.0 seems to be the best choice to balance these two tasks.

In the last part of this subsection, we make a study of the impact of level order when disigning the hierarchical structure. As we have discussed in the Datasets part in Section 4, the hierarchical structures are designed mainly based on the perception procedures by human. For the CADCars dataset, at the root level, we regards all images as the *car*. Then at the next level, we usually distinguish them based on the car type in practice, while the different poses cased by imaging angle variations in the 3D model (usually referred to “intra-class variations”) for each car type are further regarded as the finer-grained sub-categories. We can also exchange the order of these levels, as it indeed does not have the ground truth. We conduct such order reverse experiments, i.e., using the car pose firstly and car type secondly to classify car images in the hierarchical structure. We compare the hierarchical classification accuracy results (i.e., the CADCars-R in Table 1) and the semantic translation results shown in Fig. 14. In terms of classification accuracy, the average performance on real test images is quite close before and after reversing level order (0.9789 vs. 0.9798). The average accuracy on generated images does not have obvious difference (0.9731 vs. 0.9587). However, the results of generated images at each level before and after reversing the level order have some



(a) Root

(b) Level 2

(c) Level 3

(d) Level 4

Fig. 12. 2D tSNE of disentangled $\mathbf{F}_l$ by HDN without reconstruction feature loss on CelebA at different levels. For easy understanding, M and F mean male and female, S and N mean Smile and Neural, and Bl, G and Br mean Black, Golden and Brown hair, respectively.

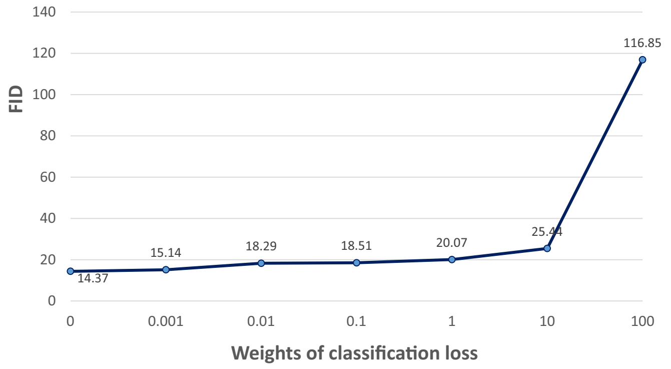


Fig. 13. FID of generated images on CelebA with different relative weight settings of the hierarchical classification loss w.r.t the adversarial loss.

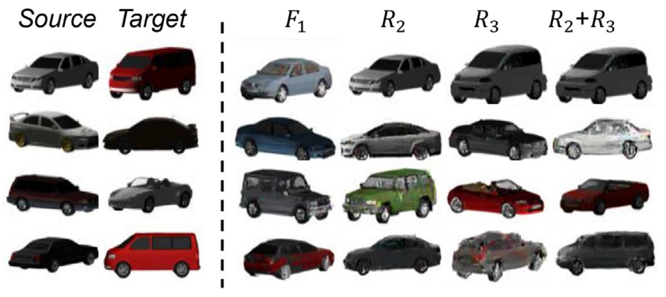


Fig. 14. Semantic translation results of the source images controlled by hierarchically disentangled features of the targets on the annotation-reversed CADCars, i.e., on this hierarchical structure, $\mathbf{R}_2$ denotes pose variation, while $\mathbf{R}_3$ denotes complex categorical variation.

difference (Lv2: 0.9792 vs. 0.9956, Lv3: 0.9670 vs. 0.9219). From the qualitative semantic translation comparisons in Figs. 6(a) and 14, we can draw similar observations, i.e., the generated images conditioned on the features involving $\mathbf{R}_3$ may have relatively poor quality. In a word, the order of the hierarchical levels may have a few impacts on the final results. When designing the hierarchical structure, following the perception procedures of human (i.e., coarse categorical change first, and then local attribute change) is better for our HDN.

5. More application scenarios

5.1. Application to image retrieval

One of the objectives of learned representations is to be applied in real-world applications. Content-based image retrieval is one of the most popular applications. Usually the retrieval objectives of users are not always clear given the query image, due to the tangled information of objects at different hierarchical levels., e.g., search the images with same category or just some same attributes and which attributes? In this part, we conduct retrieval at different levels on the predefined hierarchical CelebA data introduced in Fig. 3(a). We compare three deep hashing methods considering their space-time efficiency and competitive performance, i.e., DSH [67], HashNet [68] and SSDH [69], and the two strong pre-trained GAN methods in Section 4.2, i.e., StarGAN [63] and ELEGANT [32]. The backbones of hashing methods are same with the bottom branch of encoder E of HDN and pretrained on CASIA WebFace dataset [70]. At the l-th level, a hashing model with bit-length as same as the dimension of the concatenation of $\{\mathbf{R}_l\}_2^l$ is trained

Table 5
mAP results of retrieval for compared methods at different semantic levels. Methods with postfix of “-S” are trained with one single model supervised at the leaf-level annotations.

Level	DSH	DSH-S	HashNet	HashNet-S	SSDH	SSDH-S	StarGAN	ELEGANT-2	ELEGANT-5	HDN	HDN-B
Lv2	0.9523	0.7619	0.9483	0.8187	0.9593	0.8120	0.7474	0.9217	0.6490	0.9571	**0.9747**
Lv3	0.8010	0.5564	0.8374	0.6338	0.8445	0.6687	0.5231	0.8986	0.5659	0.8589	**0.9006**
Lv4	0.6461	0.6461	0.6336	0.6336	0.7052	0.7052	0.5016	0.4256	0.7889	0.6941	**0.7919**

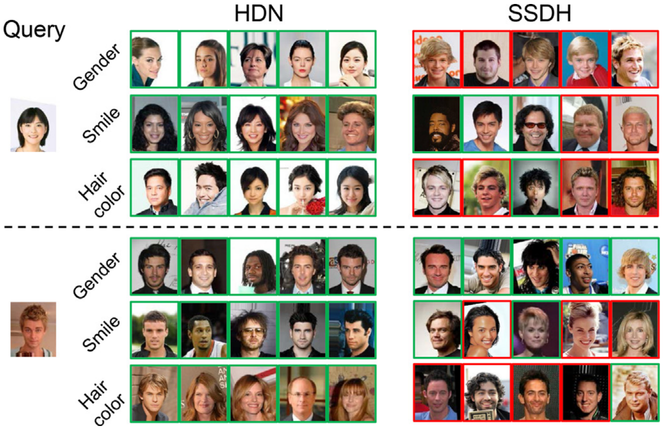


Fig. 15. Top-5 returned images of two retrieval cases using different parts of features (i.e. different $\mathbf{R}_l$ of HDN, and corresponding dimensions of bit parts of SSDH trained with leaf-level supervisions). Green and red boxes are correct and false samples respectively, judged by the hierarchical annotations (i.e., gender, smile and hair color). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

using the semantic annotations at that level. As for StarGAN and ELEGANT, the latent features before the last layer of discriminator are used as the representations of samples. To make a fair comparison with hashing methods, we also binarize $\mathbf{R}_l$ via *Sigmoid* activation during training, which we named as HDN-B. The test set are used as queries to retrieve the training set.

Table 5 gives the mean Average Precision (mAP) of retrieval evaluations at different levels. First, our method achieves the best performance, though we do not impose specific metric learning objectives on features. Besides, the compared ELEGANT variants are sometimes better than StarGAN, but are not stable, as it cannot well deal with multiple semantic concepts (more than two) when conducting the semantic features combination, limited by its flat disentangling manner. Second, HDN is more efficient since it only uses one model to handle different levels of retrieval needs owing to the disentanglement, while hashing methods have to train a model at each level. We also tried to use only one single model trained at the leaf-level, where it tangles all levels’ information in the annotations, to evaluate at high levels (methods with postfix of “-S”), but the results are inferior to those independently trained for each level. Third, HDN-B is better than HDN, which mainly due to the increased non-linear ability of features. Finally, the retrieval of HDN is more interpretable. As shown in Fig. 15, with different parts of features (i.e., different $\mathbf{R}_l$ of our HDN), the returned images satisfy different semantic requirements, while for general method like SSDH one can not interpret the meanings of different code parts, since the returned results do not present a certain semantic consistently.

5.2. Unseen category prediction and semantic edit

Discovery of unseen categories is a challenging task for deep classifier models, which has high requirements for the generalization ability of learned representations. As our HDN learns features at different hierarchical levels, it can obtain sequential category predictions for an object. Therefore, if unseen objects (e.g., a new breed of dog) share some semantic levels with seen ones, we can still obtain the right predictions at those levels (i.e., animal and dog), and the predictions at seen categories levels where unseen objects have their own unique semantics should be confused (i.e., unknown about the fine-grained dog breed). To this end, we use a linear hierarchical classifier trained with the level-wise disentangled features to evaluate the classification accuracy for levels where seen and unseen objects share the same categories, and to compute the information entropy of predictions for the level where unseen objects have their unique categorical labels. We test HDN on certain unseen leaf-level categories, i.e., bald and gray hair on CelebA, kinds of unseen tables and sofas on ShapeNet-C, and objects with other poses on ShapeNet-P, as shown in Fig. 16.

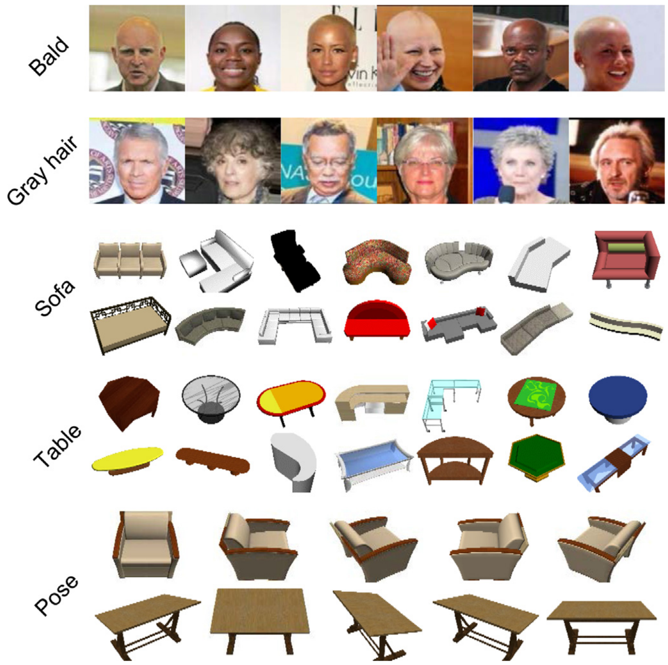


Fig. 16. Typical samples of unseen data on CelebA (bald and gray hair), ShapeNet-C (kinds of novel tables and sofas) and ShapeNet-P (more unseen poses). Large differences compared with seen hierarchical data in Figs. 3(a) and 4(b) can be found here.

Table 6
Hierarchical prediction performance for seen test set and unseen leaf-level categories. Lv2 or Lv3 included is the level where seen and unseen objects share the same semantics, and the entropy is tested at the leaf-level where unseen objects have novel semantic annotations.

	CelebA		ShapeNet-C		ShapeNet-P	
Metric	Seen	Unseen	Seen	Unseen	Seen	Unseen
Lv2 Acc.	0.9441	0.9650	1.0	0.7019	0.8563	0.7727
Lv3 Acc.	0.8520	0.9450	–	–	–	–
Entropy	0.1779	0.3561	0.1204	0.4015	0.1567	0.4913

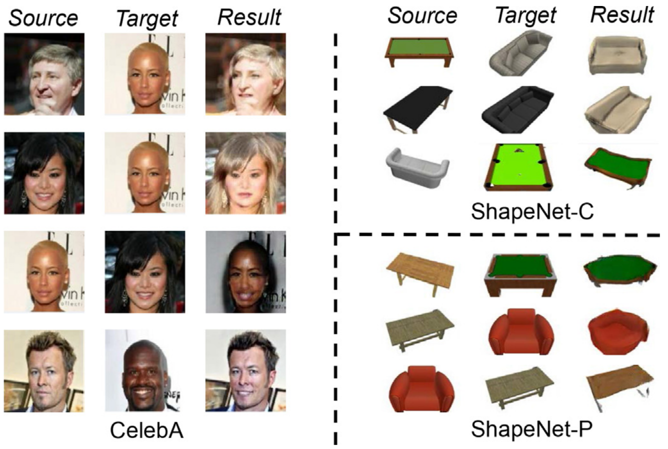


Fig. 17. Semantic translation results between seen and unseen (i.e., bald and gray hair of CelebA, kinds of novel tables and sofas of ShapeNet-C and objects with new poses of ShapeNet-P) objects. Here we replace all levels of $\mathbf{R}_l$ of the source images with those of the targets, which is equivalent to the right most case in Fig. 5.

Table 6 shows the quantitative results. Two conclusions can be reached. 1). At levels where seen and unseen objects share same semantics (i.e., levels of gender and smile on CelebA, level of Sofa/Table on ShapeNet-C, and level of Loveseat/Club chair/Work table/Billiards on ShapeNet-P), most objects can be correctly classified. 2). At 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 to be classified as appearance similar seen categories at leaf-level. For instance, about 30% and 56% bald faces are recognized as black and golden hair respectively, fifty-fifty leather couches are predicted as loveseat and L-couch respectively, and 44% and 50% of the frontal-pose sofa/table are classified as the right 30° offset of frontal and left 30° offset of frontal. The semantic translations in Fig. 17 between seen and unseen images also verify such observations. Specifically, the semantics of non-leaf levels can be extracted and transferred as usual, but the unseen unique features are not. For instance, unseen bald attribute may be disentangled as golden or black hair due to the skin color. The material of unseen leather couch is ignored on ShapeNet-C, since the trained model focuses more on shape information to distinguish seen objects rather than material information during training. The translations of tables to unseen frontal pose are also confused and dissimilar with any seen pose as shown in the cases of ShapeNet-P. Through this study, we believe that disentangling visual primitives of objects as learned knowledge is one promising solutions to the ability of open-world recognition.

6. Discussions

The proposed method in this paper builds on the framework of conditional generative adversarial network and utilize the simple hierarchical recombination scheme to learn disentangled semantic representations. It can achieve satisfactory results on relatively simple image data with clean background and clear semantic differences between levels. However, it has some limitations and needs more efforts to improve. Here we try to make further discussions on such limitations and expect to inspire more explorations in this area.

Firstly, our HDN does not perform well to handle images with complicated background and heavily tangled semantic information, such as on the ImageNet dataset shown in the supplementary materials. On the one hand, our method does not consider the geometrical relationship between semantics when recombining the disentangled features. However, given a set of semantic information (e.g. attributes), different geometrical combinations could lead to different visual perceptions. Moreover, some semantic information maybe intractable to be disentangled in the 2D space. On the other hand, currently the capacity of cGAN frameworks is limited to deal with large scale of data distributions, the training of which is not very stable. It is suggested that the usage of 3D generative models which can disentangle the intractable semantics in 2D space, or turning to recently proposed more advanced generative frameworks like the diffusion model [71] would boost the performance of our method significantly. **Secondly**, the supervised signals in current HDN depend on the human-defined hierarchical prior, which sometimes is ad-hoc. Besides, the generalization performance across datasets shown in the supplementary materials is not very well right now. In the future, one possible way to overcome these limitations is to construct an automatic self-supervised representation learning system, where the hierarchical structure is created based on the properties within data or predefined rules, such as coarse-to-fine clustering in a deep feature space. By doing so, when novel data comes, the whole system can be updated automatically.

7. Conclusions

In this paper, we propose the hierarchical disentangling network (HDN) which exploits the natural hierarchical characteristics among categories to learn the object representations in a coarse-to-fine manner (i.e., level-wise). Our model achieves promising disentangling results on several popular object image datasets. We also show the applications of such disentangled features on image-to-image translation, content-based image retrieval and even unseen objects prediction. Our current work can be viewed as an early attempt towards the long goal of disentangled representation learning, and it still has some limitations as discussed in Section 6, where we have introduced some promising future directions to improve and push the research progress in this area, such as the study of diffusion models to fit complicated data distributions, leveraging the 3D geometrical knowledge on hierarchical disentangled representation learning to avoid the ill-conditioned issues in the 2D space, and the introduction of self-supervised learning to automatically construct hierarchical prior, etc.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgment

Most of this work was done at the Institute of Computing Technology, Chinese Academy of Sciences, where Shishi Qiao pursued his Ph.D. degree. This work is partially supported by National Key R&D Program of China no. 2021ZD0111901, Natural Science Foundation of China under contracts nos. U21B2025, U19B2036, 61922080, 62206260 and Aerospace Information Research Institute of Chinese Academy of Sciences Project no. GXTC-A1-21630167.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.patcog.2023.109539.

References

[1] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M.S. Bernstein, A.C. Berg, F. Li, Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[2] S. Ren, K. He, R.B. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: NIPS, 2015, pp. 91–99.

[3] J. Redmon, S.K. Divvala, R.B. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: IEEE, CVPR, 2016, pp. 779–788.

[4] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S.E. Reed, C. Fu, A.C. Berg, SSD: single shot multibox detector, in: ECCV, 2016, pp. 21–37.

[5] I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A.C. Courville, Y. Bengio, Generative adversarial nets, in: NIPS, 2014, pp. 2672–2680.

[6] M.D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: ECCV, 2014, pp. 818–833.

[7] A. Dosovitskiy, T. Brox, Inverting visual representations with convolutional networks, in: IEEE, CVPR, 2016, pp. 4829–4837.

[8] D. Bau, B. Zhou, A. Khosla, A. Oliva, A. Torralba, Network dissection: quantifying interpretability of deep visual representations, in: IEEE, CVPR, 2017, pp. 3319–3327.

[9] K. Simonyan, A. Vedaldi, A. Zisserman, Deep inside convolutional networks: visualising image classification models and saliency maps, ICLR, 2014.

[10] P. Stock, M. Cissé, Convnets and imagenet beyond accuracy: explanations, bias detection, adversarial examples and model criticism, CoRRabs/1711.11443 (2017).

[11] Q. Zhang, Y.N. Wu, S. Zhu, Interpretable convolutional neural networks, in: IEEE, CVPR, 2018, pp. 8827–8836.

[12] S.E. Reed, K. Sohn, Y. Zhang, H. Lee, Learning to disentangle factors of variation with manifold interaction, in: ICML, 2014, pp. 1431–1439.

[13] M. Mathieu, J.J. Zhao, P. Sprechmann, A. Ramesh, Y. LeCun, Disentangling factors of variation in deep representation using adversarial training, in: NIPS, 2016, pp. 5041–5049.

[14] S. Rifai, Y. Bengio, A.C. Courville, P. Vincent, M. Mirza, Disentangling factors of variation for facial expression recognition, in: ECCV, 2012, pp. 808–822.

[15] L. Tran, X. Yin, X. Liu, Disentangled representation learning GAN for pose-invariant face recognition, in: IEEE,CVPR, 2017, pp. 1283–1292.

[16] A. Gonzalez-Garcia, J. van de Weijer, Y. Bengio, Image-to-image translation for cross-domain disentanglement, in: NIPS, 2018, pp. 1294–1305.

[17] X. Huang, M. Liu, S.J. Belongie, J. Kautz, Multimodal unsupervised image-to-image translation, in: ECCV, 2018, pp. 179–196.

[18] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, P. Abbeel, InfoGAN: interpretable representation learning by information maximizing generative adversarial nets, in: NIPS, 2016, pp. 2172–2180.

[19] J. Xie, Y. Xu, E. Nijkamp, Y.N. Wu, S. Zhu, Generative hierarchical learning of sparse FRAME models, in: IEEE, CVPR, 2017, pp. 1933–1941.

[20] S. Zhao, J. Song, S. Ermon, Learning hierarchical features from deep generative models, in: ICML, 2017, pp. 4091–4099.

[21] Y. Bengio, A.C. Courville, P. Vincent, Representation learning: a review and new perspectives, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1798–1828.

[22] M. Mirza, S. Osindero, Conditional generative adversarial nets, CoRRabs/1411.1784 (2014).

[23] J.B. Tenenbaum, W.T. Freeman, Separating style and content, in: NIPS, 1996, pp. 662–668.

[24] C. Wang, C. Wang, C. Xu, D. Tao, Tag disentangled generative adversarial network for object image re-rendering, in: IJCAI, 2017, pp. 2901–2907.

[25] B. Tong, C. Wang, M. Klinkigt, Y. Kobayashi, Y. Nonaka, Hierarchical disentanglement of discriminative latent features for zero-shot learning, in: IEEE, CVPR, 2019, pp. 11467–11476.

[26] T. Kaneko, K. Hiramatsu, K. Kashino, Generative adversarial image synthesis with decision tree latent controller, in: IEEE, CVPR, 2018, pp. 6606–6615.

[27] Y. Alharbi, P. Wonka, Disentangled image generation through structured noise injection, in: IEEE, CVPR, 2020, pp. 5134–5142.

[28] Y. Deng, J. Yang, D. Chen, F. Wen, X. Tong, Disentangled and controllable face image generation via 3D imitative-contrastive learning, in: IEEE, CVPR, 2020, pp. 5154–5163.

[29] Y. Shen, C. Yang, X. Tang, B. Zhou, InterFaceGAN: interpreting the disentangled face representation learned by GANs, IEEE Trans. Pattern Anal. Mach. Intell. 44 (4) (2022) 2004–2018.

[30] J. Mu, S.D. Mello, Z. Yu, N. Vasconcelos, X. Wang, J. Kautz, S. Liu, CoordGAN: self-supervised dense correspondences emerge from GANs, IEEE, CVPR, 2022.

[31] K. Wadhwani, S.P. Awate, Controllable image generation with semi-supervised deep learning and deformable-mean-template based geometry-appearance disentanglement, Pattern Recognit. 118 (2021) 108001.

[32] T. Xiao, J. Hong, J. Ma, ELEGANT: exchanging latent encodings with GAN for transferring multiple face attributes, in: ECCV, 2018, pp. 172–187.

[33] A. Dosovitskiy, T. Brox, Generating images with perceptual similarity metrics based on deep networks, in: NIPS, 2016, pp. 658–666.

[34] D. Bau, J. Zhu, H. Strobelt, B. Zhou, J.B. Tenenbaum, W.T. Freeman, A. Torralba, GAN dissection: visualizing and understanding generative adversarial networks, ICLR, 2019.

[35] R.C. Fong, A. Vedaldi, Interpretable explanations of black boxes by meaningful perturbation, in: IEEE, ICCV, 2017, pp. 3449–3457.

[36] L.M. Zintgraf, T.S. Cohen, T. Adel, M. Welling, Visualizing deep neural network decisions: prediction difference analysis, ICLR, 2017.

[37] R. Abbasi-Asl, B. Yu, Interpreting convolutional neural networks through compression, CoRRabs/1711.02329 (2017).

[38] S. Palacio, J. Folz, J. Hees, F. Raue, D. Borth, A. Dengel, What do deep networks like to see? in: IEEE, CVPR, 2018, pp. 3108–3117.

[39] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F.A. Wichmann, W. Brendel, Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness, ICLR, 2019.

[40] O. Li, H. Liu, C. Chen, C. Rudin, Deep learning for case-based reasoning through prototypes: a neural network that explains its predictions, in: AAAI, 2018, pp. 3530–3537.

[41] L. Kook, L. Herzog, T. Hothorn, O. Dürr, B. Sick, Deep and interpretable regression models for ordinal outcomes, Pattern Recognit. 122 (2022) 108263.

[42] G. Griffin, P. Perona, Learning and using taxonomies for fast visual categorization, in: IEEE, CVPR, 2008, pp. 1–8.

[43] M. Marszalek, C. Schmid, Constructing category hierarchies for visual recognition, in: ECCV, 2008, pp. 479–491.

[44] J. Deng, J. Krause, A.C. Berg, F. Li, Hedging your bets: optimizing accuracy-specificity trade-offs in large scale visual recognition, in: IEEE, CVPR, 2012, pp. 3450–3457.

[45] V. Ordonez, J. Deng, Y. Choi, A.C. Berg, T.L. Berg, From large scale image categorization to entry-level categories, in: IEEE, ICCV, 2013, pp. 2768–2775.

[46] J. Deng, N. Ding, Y. Jia, A. Frome, K. Murphy, S. Bengio, Y. Li, H. Neven, H. Adam, Large-scale object classification using label relation graphs, in: ECCV, 2014, pp. 48–64.

[47] N. Ding, J. Deng, K.P. Murphy, H. Neven, Probabilistic label relation graphs with Ising models, in: IEEE, ICCV, 2015, pp. 1161–1169.

[48] B. Zhao, F. Li, E.P. Xing, Large-scale category structure aware image categorization, in: NIPS, 2011, pp. 1251–1259.

[49] N. Srivastava, R. Salakhutdinov, Discriminative transfer learning with tree-based priors, in: NIPS, 2013, pp. 2094–2102.

[50] S.J. Hwang, L. Sigal, A unified semantic embedding: relating taxonomies and attributes, in: NIPS, 2014, pp. 271–279.

[51] Z. Yan, H. Zhang, R. Piramuthu, V. Jagadeesh, D. DeCoste, W. Di, Y. Yu, HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition, in: IEEE, ICCV, 2015, pp. 2740–2748.

[52] W. Goo, J. Kim, G. Kim, S.J. Hwang, Taxonomy-regularized semantic deep convolutional neural networks, in: ECCV, 2016, pp. 86–101.

[53] K. Ahmed, M.H. Baig, L. Torresani, Network of experts for large-scale image categorization, in: ECCV, 2016, pp. 516–532.

[54] K.K. Singh, U. Ojha, Y.J. Lee, FineGAN: unsupervised hierarchical disentanglement for fine-grained object generation and discovery, in: IEEE, CVPR, 2019, pp. 6490–6499.

[55] X. Mao, Q. Li, H. Xie, R.Y.K. Lau, Z. Wang, S.P. Smolley, Least squares generative adversarial networks, in: IEEE, ICCV, 2017, pp. 2813–2821.

[56] T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, in: IEEE, CVPR, 2019, pp. 4401–4410.

[57] X. Huang, S.J. Belongie, Arbitrary style transfer in real-time with adaptive instance normalization, in: IEEE, ICCV, 2017, pp. 1510–1519.

[58] Z. Liu, P. Luo, X. Wang, X. Tang, Deep learning face attributes in the wild, in: IEEE, ICCV, 2015, pp. 3730–3738.

[59] H. Xiao, K. Rasul, R. Vollgraf, Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms, CoRRabs/1708.07747 (2017).

[60] S. Fidler, S.J. Dickinson, R. Urtasun, 3D object detection and viewpoint estimation with a deformable 3D cuboid model, in: NIPS, 2012, pp. 620–628.

[61] A.X. Chang, T.A. Funkhouser, L.J. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, J. Xiao, L. Yi, F. Yu, ShapeNet: an information-rich 3D model repository, CoRRabs/1512.03012 (2015).

[62] L.v.d. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 9 (Nov) (2008) 2579–2605.

[63] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, J. Choo, StarGAN: unified generative adversarial networks for multi-domain image-to-image translation, in: IEEE, CVPR, 2018, pp. 8789–8797.

[64] T. Salimans, I.J. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, Improved techniques for training GANs, in: NIPS, 2016, pp. 2226–2234.

[65] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, GANs trained by a two time-scale update rule converge to a local Nash equilibrium, in: NIPS, 2017, pp. 6626–6637.

[66] R. Zhang, P. Isola, A.A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, in: IEEE, CVPR, 2018, pp. 586–595.

[67] H. Liu, R. Wang, S. Shan, X. Chen, Deep supervised hashing for fast image retrieval, in: IEEE, CVPR, 2016, pp. 2064–2072.

[68] Z. Cao, M. Long, J. Wang, P.S. Yu, HashNet: deep learning to hash by continuation, in: IEEE, ICCV, 2017, pp. 5609–5618.

[69] H. Yang, K. Lin, C. Chen, Supervised learning of semantics-preserving hash via deep convolutional neural networks, IEEE Trans. Pattern Anal. Mach. Intell. 40 (2) (2018) 437–451.

[70] D. Yi, Z. Lei, S. Liao, S.Z. Li, Learning face representation from scratch, CoRRabs/1411.7923 (2014).

[71] J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, in: NIPS, 2020, pp. 6840–6851.

Shishi Qiao received the B.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2014, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2021. In 2021, he joined the Faculty of the Department of Information Science and Engineering, Ocean University of China (OUC), Qingdao, China, where he has been an Assistant Professor. His research interests mainly include computer vision, pattern recognition, machine learning and, in particular, video face recognition, multimedia retrieval, object and scene understanding with deep generative models.

Ruiping Wang received the B.S. degree in applied mathematics from Beijing Jiaotong University, Beijing, China, in 2003, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, in 2010. He was a Post Doctoral Researcher with the Department of Automation, Tsinghua University, Beijing, from 2010 to 2012. He also spent one year as a Research Associate with the Computer Vision Laboratory, Institute for Advanced Computer Studies, University of Maryland at College Park, College Park, from 2010 to 2011. In 2012, he joined the Faculty of the Institute of Computing Technology, Chinese Academy of Sciences, where he has been a Professor since 2017. His research interests include computer vision, pattern recognition, and machine learning.

Shiguang Shan received the M.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 1999, and the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2004. In 2002, he joined ICT, CAS, where he has been a Professor since 2010. He is currently the Deputy Director of the Key Laboratory of Intelligent Information Processing, CAS. He has authored over 200 papers in refereed journals and proceedings in computer vision and pattern recognition. His research interests include computer vision, pattern recognition, and machine learning. He especially focuses on face recognition related research topics. He was a recipient of the Chinas State Natural Science Award in 2015 and the Chinas State S&T Progress Award in 2005 for his research work. He is an Associate Editor of several journals, including the IEEE TRANSACTIONS ON IMAGE PROCESSING, the Computer Vision and Image Understanding, the Neurocomputing, and the Pattern Recognition Letters. He has served as the Area Chair for some international conferences, including ICCV11, ICPR12/14/20, ACCV12/16/18, FG13/18/20, ICASSP14, BTAS18, and CVPR19/20.

Xilin Chen is a professor with the Institute of Computing Technology, Chinese Academy of Sciences (CAS). He has authored one book and more than 300 papers in refereed journals and proceedings in the areas of computer vision, pattern recognition, image processing, and multimodal interfaces. He is currently an information sciences editorial board member of Fundamental Research, an editorial board member of Research, a senior editor of the Journal of Visual Communication and Image Representation, and an associate editor-in-chief of the Chinese Journal of Computers, and Chinese Journal of Pattern Recognition and Artificial Intelligence. He served as an organizing committee member for multiple conferences, including general co-chair of FG 2013 / FG 2018, VCIP 2022, and program co-chair of ICMI 2010. He is a fellow of the ACM, IEEE, IAPR, and CCF.