iCaps: An Interpretable Classifier via Disentangled Capsule Networks

Dahuin Jung, Jonghyun Lee, Jihun Yi, and Sungroh Yoon*

Electrical and Computer Engineering,
ASRI, INMC, and Institute of Engineering Research
Seoul National University, Seoul 08826, South Korea
{annajung0625, leejh9611, t080205, sryoon}@snu.ac.kr

Abstract. We propose an interpretable Capsule Network, *iCaps*, for image classification. A capsule is a group of neurons nested inside each layer, and the one in the last layer is called a class capsule, which is a vector whose norm indicates a predicted probability for the class. Using the class capsule, existing Capsule Networks already provide some level of interpretability. However, there are two limitations which degrade its interpretability: 1) the class capsule also includes classification-irrelevant information, and 2) entities represented by the class capsule overlap. In this work, we address these two limitations using a novel class-supervised disentanglement algorithm and an additional regularizer, respectively. Through quantitative and qualitative evaluations on three datasets, we demonstrate that the resulting classifier, *iCaps*, provides a prediction along with clear rationales behind it with no performance degradation.

Keywords: Capsule Networks, Interpretable Neural Networks, Class-supervised Disentanglement, Generative Adversarial Networks (GANs)

1 Introduction

Despite the success of deep learning in a broad range of tasks, including image classification and segmentation, speech synthesis, and medical decision-making, the reliability of decisions made by artificial intelligence is still questionable. Hence, many promising studies have been conducted regarding explainable artificial intelligence (XAI). The main task of XAI is to provide explanations that can aid the comprehension of provided decisions to users. Using these explanations, users can check whether a model performs as expected or identify potential bias/problems inherent to the model.

Several different approaches have been proposed to explain deep learning models. In some studies, models that can provide human-understandable explanations of their predictions without retraining or modification have been proposed. These studies aim for built-in interpretability. We herein propose a new built-in interpretable model that offers a concept-based explanation using Capsule Networks (CapsNets) [38].

* Correspondence to: Sungroh Yoon sryoon@snu.ac.kr

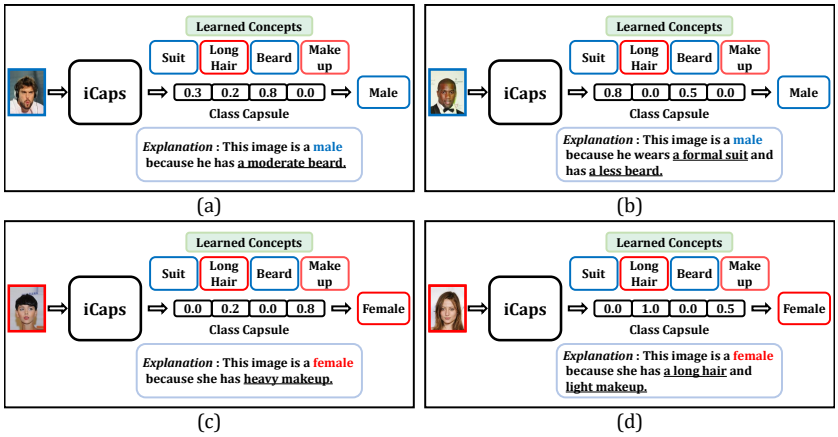

Fig. 1. Overview of our study. We propose a new interpretable classifier, iCaps, which classifies an observation by only considering class-relevant variables; these class-relevant variables are human-understandable concepts. By analyzing the values of the class-relevant variables (concepts), we can understand the decisions made by iCaps.

As the main building block, CapsNets use capsules - a group of neurons – that encapsulates the instantiation parameters of an entity, such as an object or its fragments. The magnitude of the output vector of a capsule indicates the probability that the encoded instantiation parameter is present in the input. The capsules in the final layer are called class capsules, and the norm of each class capsule indicates the predicted probability of each class. The instantiation parameters of the class capsule can represent the position, color, texture, and scaling of an object or its fragments, and these can be interpreted as concepts to humans.

Therefore, the instantiation parameters represented by the class capsule and its magnitudes can be used, to an extent, to explain a model's prediction. However, two factors degrade interpretability. First, some instantiation parameters of the class capsule represent classification-irrelevant concepts. Next, a single concept can be encoded in multiple elements of the class capsule. Therefore, a single concept can be represented by different magnitudes in two different elements.

By addressing these two problems, we propose an interpretable CapsNet architecture, *iCaps*, that only contains classification-relevant distinct concepts in the class capsule. The overview of iCaps is described in Fig. 1. To address the first problem, we propose a novel class-supervised disentanglement method that disentangles class-relevant and -irrelevant features within an observation, without any leakage. For the second problem, we use an additional regularizer based on latent traversal to prevent the same concept from being encapsulated several times in the class capsule.

Some built-in interpretable models use predefined concepts to provide explanations [22,21]. However, such prior knowledge is not available or costly to define

in most cases; hence, in this study, we assumed where the concepts were learned instead. Moreover, we posit three desiderata for an interpretable classifier based on the learned concepts: informativeness, distinctness, and explainability where, for example, informativeness ensures that only classification-relevant information is used to provide an explanation of the model's prediction. Based on the three desiderata, we validated our model both theoretically and empirically. Our main contributions in this study are as follows:

1. We improve the explainability property of CapsNets by addressing two problems: classification-irrelevant information and overlapping.
2. We suggest a novel class-disentanglement algorithm that can disentangle the latent feature of x into two complementary subspaces: class-relevant and -irrelevant subspaces. The class-relevant subspace of our algorithm contains intra-class variation, unlike prior studies, which contain only inter-class variation.
3. We posit three desiderata for an interpretable classifier based on learned concepts and demonstrate the effectiveness of iCaps based on the three desiderata.

2 Related Work

2.1 Capsule Networks

CapsNet [38] is a neural network based on a group of neurons – a vector. The original CapsNet has a simple network structure comprising three layers. First, an observation x undergoes a convolution layer to transfer pixel-level information into a latent space, followed by the Primary-Capsule (PC) layer and Class-Capsule (CC) layer. Information contained in the PC layer is transferred to the CC layer above using a dynamic routing method, and this method is called "routing by agreement". The coupling coefficients between the capsules in the PC and CC layers are updated in a direction that can increase the classification performance (a top-down mechanism). The output of the CC layer is a class capsule of classes, and the norm of each class capsule indicates the predicted probability for each class. The elements of the class capsules represent the instantiation parameters of a type of entity. To encode these instantiation parameters for the class capsules, margin loss and reconstruction loss are used. The margin loss, $\mathcal{L}_M$, is:

$$\mathcal{L}_M = \lambda_M \left(y^i \max(0,\, m^+ - \|c\|)^2 + 0.5\, ((1 - y^i) \max(0,\, \left\|c^{\neq i}\right\| - m^-)^2)\right), \quad (1)$$

where $y^i = 1$ iff the ground-truth class label is i, c is the class capsule for the ground-truth class i, $c^{\neq i}$ are the class capsules except for c, $m^+ = 0.9$, and m^- $= 0.1$. The reconstruction loss, $\mathcal{L}_{\text{recon}}$, is used, which is expressed as

$$\mathcal{L}_{\text{recon}} = \lambda_{\text{recon}}\, \mathbb{E}\, \|\hat{x} - x\|_F^2\,, \quad (2)$$

where $\hat{x}$ is reconstructed x using an additional decoder, which uses c as the input. The original CapsNet presents some computational and structural limitations.

Hence, some advanced studies based on CapsNet have been performed [2,15,19,36]. Among these, our study uses DeepCaps [36] as a base because it yields better classification performance than the original CapsNet on more complex images by utilizing a skip connection and a three-dimensional convolution-inspired routing method and is easy to apply to our work. More detailed information regarding CapsNets is available in [38,36].

2.2 Disentanglement

Our work utilizes class-supervised disentanglement to create an interpretable classifier that provides an explanation using only classification-relevant information. Class-supervised disentanglement learning aims to disentangle the latent feature of x into two complementary subspaces - class-relevant and -irrelevant subspaces - in a setting where the class label for images in the training set is provided. Two approaches can be used in class-supervised disentanglement: adversarial and non-adversarial. DrNet [9] and Szabo et al. [46] are adversarial methods, whereas Cycle-VAE [20], ML-VAE [4], and LORD [12] are non-adversarial methods. Implicitly or explicitly, all class-supervised disentanglement methods assume that inter-class variation is much larger than intra-class variation; therefore, intra-class variation can be ignored. On the contrary, our work assumes that intra-class variation should not be ignored even though it is relatively small. From the perspective of an interpretable classifier, intra-class variation is an important feature to explain a model's prediction. Also, our method has some level of similarity with InfoGAN [7] (unsupervised disentanglement method) in a structural way. The comparison is given in Sec. S10 of the supplementary.

2.3 Interpretable Methods

Two topics of research in XAI provide two different notions of interpretability of deep models: (1) post-hoc interpretability and (2) built-in interpretability. (1) Post-hoc interpretability methods aim to interpret models or decisions of already trained neural networks. By contrast, networks that are inherently interpretable provide (2) built-in interpretability.

Post-hoc Interpretability Starting from the Saliency map [42], a number of post-hoc interpretation methods have been suggested to visually explain the decision of a classifier. These methods generate a heatmap of the same size as the input image and highlight the decisive regions within the input image. Post-hoc methods are based on backpropagation [1,3,40,41,42,43,44,45], local perturbation [47,48], or mask optimization [5,8,10,11,34]. The resulting heatmap of post-hoc methods are easily interpretable. However, evaluating the quality of their results is non-trivial, and both the method and the evaluation metrics are active research areas. Recently, several interpretable methods that offer explanations based on concepts and prototypes have been proposed [22,31,27,6]. TCAV [22] is a post-hoc method based on predefined concepts. TCAV offers an explanation by

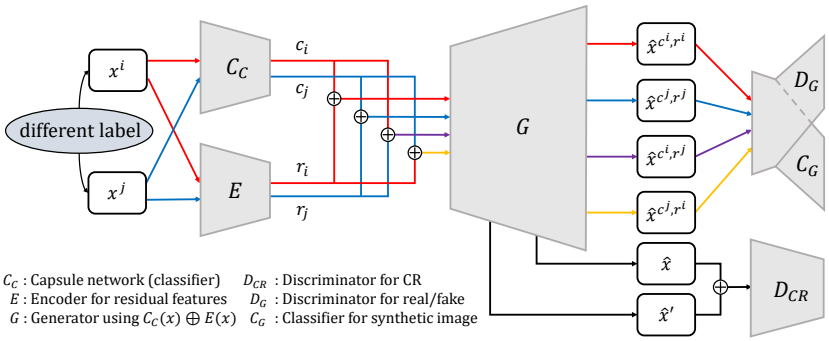

Fig. 2. Network architecture. x^i and x^j are two images of different labels. C_C encodes only class-relevant features of x^i and x^j to c^i and c^j, respectively. E encodes only class-irrelevant (residual) features of x^i and x^j to r^i and r^j, respectively. G constructs the images $\hat{x}^{c^i,r^i}$ and $\hat{x}^{c^j,r^j}$ using $c^i \oplus r^i$ and $c^j \oplus r^j$, respectively. Also, G generates the images $\hat{x}^{c^i,r^j}$ and $\hat{x}^{c^j,r^i}$ using swapped $c^i \oplus r^j$ and $c^i \oplus r^j$, respectively. These images are distinguished as real or fake and classified by D_G and C_G. D_{CR} takes two images $\hat{x}$ and $\hat{x}'$ that share the same c and r except for a single index l of c, and it is trained to identify l.

finding the closest predefined concepts to the corresponding class in the feature space. Unlike most post-hoc methods, TCAV only offers an explanation for a class, not for a single data point.

Built-in Interpretability SENN [31] suggests an interpretable classifier structure that predicts a class by combining concepts and relevances. SENN encodes input features into two representations: concepts and relevances. The importance of a given concept for classification can be explained through the relevance score of the corresponding concept. However, the learned concepts of SENN are not clearly human-understandable, as analyzed in Sec. S3 of the supplementary. ProtoPNet [6] proposes an interpretable classifier based on prototypes. ProtoPNet learns prototypical patches of each class from the training dataset. After finding the prototypes, the model makes a decision by measuring the distance between local patches of the test observation and the found prototypes of each class.

3 iCaps: An Interpretable Classifier via Disentangled Capsule Networks

iCaps comprises six components, as illustrated in Fig. 2.

- C_C: a capsule network (classifier) that represents the class-relevant latent space.
- E: an encoder that represents the class-irrelevant (residual) latent space.

- G: a generator that creates synthetic images using $C_C(x) \oplus E(x)$, where $\oplus$ represents concatenation.
- D_G: a discriminator for image generation, that distinguishes whether an observation is from the dataset or from G.
- C_G: a classifier for image generation, that estimates class labels.
- D_{CR}: a discriminator for contrastive regularization (CR), that maximizes the distance between the concepts represented by C_C

Assume that a collection of n images $x_1, x_2, ..., x_n \in \mathcal{X}$ and their corresponding labels $y_i \in [k]$ is provided. k and $[k]$ (=[1,...,k]) are the number and the set of classes, respectively. $\mathcal{X}^i$ represents all images corresponding to a class index i. As described in Fig. 2, iCaps uses two images of different class labels as input in the training phase. In the case of binary classification, the input pairs are images of the opposite class labels. In multiclass classification, two class labels are randomly selected in each batch.

We assume that the representation of images can be disentangled into two complementary latent spaces, $\mathcal{C}$ and $\mathcal{R}$. Our objective is to find a class-relevant representation $c_i \in \mathcal{C}$ and a class-irrelevant (we call as residual) representation $r_i \in \mathcal{R}$ for each image x_i. The size of output vector of the class-relevant representation is L.

3.1 Disentanglement between Class-relevant and Class-irrelevant Information

The class-relevant subspace, $\mathcal{C}$, is represented by C_C, and the residual subspace, $\mathcal{R}$, is represented by E. The class-relevant representation, c_i, contains all the information relevant for classification, whereas the residual representation, r_i, contains residual information irrelevant for classification. The previous class-supervised disentanglement methods [9,20,4,12] contain only information shared by each class (inter-class variation) in c_i, assuming that $\|c_i - c_j\|_F^2 = 0$ if $y_i = y_j$. However, under this assumption, c_i cannot include classification-relevant intra-class variation. For example, if a model is trained using a dataset labeled as female and male, and most of the males are wearing suits, then wearing a suit should be a classification-relevant variable. In other words, this variable should be included in c. However, to satisfy the assumption: $\|c_i - c_j\|_F^2 = 0$ if $y_i = y_j$ above, every man should be defined as wearing a suit. This is not true. If this variable is not included in c, it should be included in r. Consequently, information leakage is implied because wearing a suit is classification-relevant information in this model.

We wish to include classification-relevant intra-class variation in c_i. By analyzing only c_i, we can understand the rationale behind the model's prediction. More information-theoretically, when the mutual information between c and y is:

$$I(c;\, y) = \int_y \int_c p(y)\, q(c|y) \log \frac{q(c|y)}{q(c)}\, dc\, dy, \tag{3}$$

where $q(c) = \int_y p(y)\, q(c|y)\, dy$, $I(c;\, y)$ should be non-zero, and $I(r;\, y)$ should be zero. To obtain $I(c;\, y) > 0$ and $I(r;\, y) = 0$, we utilize three objective functions.

The first objective is a cross-entropy loss that includes two images generated by swapping r of two inputs. The loss term is:

$$\begin{aligned}\mathcal{L}_{C_G} = \lambda_{C_G}\, (&\mathbb{E}[(1 - y_t) \cdot C_G(x)] + \mathbb{E}[(1 - y_t^i) \cdot C_G(\hat{x}^{c^i, r^i})] + \\ &\mathbb{E}[(1 - y_t^i) \cdot C_G(\hat{x}^{c^i, r^j})] - \mathbb{E}[y_t \cdot C_G(x)]] - \\ &\mathbb{E}[y_t^i \cdot C_G(\hat{x}^{c^i, r^i})]] - \mathbb{E}[y_t^i \cdot C_G(\hat{x}^{c^i, r^j})]]),\end{aligned} \tag{4}$$

where i and j are two different class indices, $x \sim p_{\text{data}}(x)$, $x^i \sim p_{\text{data}}(x^i)$, $x^j \sim p_{\text{data}}(x^j)$, y_t is an one-hot encoding of $y \in [k]$, $\hat{x}^{c^i, r^i} \sim G(C_C(x^i), E(x^i))$, and $\hat{x}^{c^i, r^j} \sim G(C_C(x^i), E(x^j))$. The class labels of the two synthetic images, $\hat{x}^{c^i, r^i}$ and $\hat{x}^{c^i, r^j}$, are the same as y_t^i. For all images in Eq. 4, the higher the probability of class i, the smaller the loss.

Furthermore, we use the prediction confidence of the model as a loss. We propose a class-similarity (CS) loss:

$$\mathcal{L}_{\text{CS}} = \lambda_{\text{CS}}\, \mathbb{E} \left\| C_G^{\text{logit}}(\hat{x}^{c^i, r^i}) - C_G^{\text{logit}}(\hat{x}^{c^i, r^j}) \right\|_F^2, \tag{5}$$

where C_G^{logit} represents the logit of C_G. In Eq. 5, we can compare the likelihood distributions of two images generated based on the same c yet different r. That is, the images generated with the same c should have exactly the same likelihood distributions in our framework. $\mathcal{L}_{C_G}$ and $\mathcal{L}_{\text{CS}}$ cause information relevant to the classification to be included in c.

The third loss is for the residual features independent of classification. Other similar studies use either an adversarial loss [9], KL-divergence term [4], or asymmetric noise regularization [12] to encode class-irrelevant features in r. Unlike these methods, we allow r to include the remaining information by causing c to include all the relevant information for classification. The residual similarity (RS) loss is defined as follows:

$$\mathcal{L}_{\text{RS}} = \lambda_{\text{RS}}\, \mathbb{E} \left\| E(x^i) - E(\hat{x}^{c^j, r^i}) \right\|_F^2. \tag{6}$$

To minimize $\mathcal{L}_{\text{RS}}$, the class-irrelevant (residual) feature, r, of the real image x^i corresponding to class index i and the class-irrelevant (residual) feature, r, of $\hat{x}^{c^j, r^i}$ should be the same. This is only possible when c contains all the classification-relevant information.

In addition, in our study, we assume that all dimensions of c are used to represent class-relevant concepts. More formally,

$$\min_{l \in L} \mathbb{E}[I(C_C(x); y)_l] > 0 \tag{7}$$

where L is the size of c. However, $\mathcal{L}_M$, $\mathcal{L}_{C_G}$ and $\mathcal{L}_{\text{CS}}$ do not guarantee the above assumption. Therefore, we propose the loss as follows:

$$\mathcal{L}_{\text{concept}} = -\, \lambda_{\text{concept}}\, (\min_{l \in L} \left| \mathbb{E}[C_C(x^i)] - \mathbb{E}[C_C(x^j)] \right|_l). \tag{8}$$

$\mathcal{L}_{\text{concept}}$ is additionally suggested to maximize the mutual information between all elements of c and y. By using $\mathcal{L}_M$ and $\mathcal{L}_{\text{concept}}$ together, we enforce all dimensions of c to represent discriminative concepts by maximizing the smallest distance between different classes i, j among all the dimensions. We elaborate the benefits of the proposed losses and components in Secs. 4 and S2.

3.2 Latent Traversal

We add a discriminator, D_{CR}, to prevent overlapping concepts, which is the second problem we wish to solve. The idea of D_{CR} comes from latent traversal. Latent traversal refers to generating images by traversing a single element of a latent space; it is widely used to measure disentanglement in evaluation [18,23]. Lin et al. [28] first used the idea of latent traversal in the training phase, and named as contrastive regularization. By following it, we also call our loss as $\mathcal{L}_{\text{CR}}$. $\mathcal{L}_{\text{CR}}$ can be expressed as follows:

$$\mathcal{L}_{\text{CR}} = -\lambda_{\text{CR}}\ \mathbb{E}_{l\sim U[L],\,(\hat{x},\hat{x}')\sim G(C_C(x),E(x))}\left[\langle I,\ \log D_{\text{CR}}(\hat{x},\hat{x}')\rangle\right], \tag{9}$$

where l is a random index over L, $\hat{x}$ and $\hat{x}'$ are two images generated with different value of $C_C(x)_l$ while fixing the remaining elements. $< \cdot >$ represents a dot product, and I denotes the one-hot encoding of the random index l. $\mathcal{L}_{\text{CR}}$ forces changes in the elements of c to be visually noticeable and easy to distinguish between each other. The difference between our L_{CR} and that reported in [28] is that we directly used the definition of latent traversal. However, Lin et al. [28] fixed a single element and changes all the remaining elements, and tries to identify the fixed element.

3.3 Interpretable Classifier Based on Learned Concepts

In addition, we replace the decoder part of DeepCaps with Generative Adversarial Networks (GANs) [13] to encourage $\hat{x}^{c^i,r^j}$ to be realistic. To enable $\mathcal{L}_{C_G}$ and $\mathcal{L}_{\text{CS}}$ to function as intended, the quality of generated image $\hat{x}^{c^i,r^j}$ is important. Unlike $\hat{x}^{c^i,r^i}$, $\hat{x}^{c^i,r^j}$ does not have a ground-truth image. Therefore, we used an adversarial game of GANs and the ACGAN [33] structure; as such, C_G and D_G share the weight except for the last fully connected layer. The losses for G and D_G are as follows:

$$\mathcal{L}_G = -\lambda_G\,\mathbb{E}[D_G(\hat{x})],\ \ \mathcal{L}_{D_G} = \lambda_{D_G}\,(\mathbb{E}[D_G(\hat{x})] - \mathbb{E}[D_G(x)]). \tag{10}$$

L_G is a loss to create a realistic image, and L_{D_G} is a WGAN [14] based loss to distinguish generated images by G from real images. For the gradient penalty, we used a Lipschitz gradient penalty term [35]:

$$\mathcal{L}_{\text{LGP}} = \lambda_{\text{LGP}}\,\mathbb{E}(\|\nabla_{\hat{x}} D_G(\hat{x})\|_2 - 1)^2_+. \tag{11}$$

The built-in interpretable model domain is still new. By analyzing similar studies, we posit a reasonable set of desiderata for an interpretable classifier based on learned concepts as follows:

Table 1. Classification accuracy (C_C) of $p(y|c)$ (higher is better).

Architecture	MNIST	SVHN	CelebA
ResNet-18 [16]	0.992	0.945	0.977
DeepCaps [36]	0.997	0.971	0.974
Ours	0.992	0.920	0.984

1. **Informativeness**: the concept representation of x for explanations should preserve only classification-relevant information,
2. **Distinctness**: the learned concepts should be non-overlapping,
3. **Explainability**: a decision should be explained with human-understandable concepts.

We obtained these conditions by (i) encoding only the class-relevant information in c, (ii) enforcing distinctness by an additional discriminator and (iii) exploiting the fact that the instantiation parameters represented by the class capsule are human-understandable concepts.

To train iCaps, we alternatively trained C_C, E, D_G & C_G, G, and D_{CR} using the following gradients:

$$\theta_{C_C} \overset{+}{\leftarrow} -\Delta_{\theta_{C_C}}(\mathcal{L}_M + \mathcal{L}_{\mathrm{recon}} + \mathcal{L}_{\mathrm{concept}} + \mathcal{L}_{C_G} + \mathcal{L}_{\mathrm{CS}} + \mathcal{L}_{\mathrm{RS}} + \mathcal{L}_{\mathrm{CR}}) \tag{12}$$

$$\theta_E \overset{+}{\leftarrow} -\Delta_{\theta_E}(\mathcal{L}_{\mathrm{recon}} + \mathcal{L}_{\mathrm{KL}} + \mathcal{L}_{\mathrm{CS}} + \mathcal{L}_{\mathrm{RS}} + \mathcal{L}_{C_G}) \tag{13}$$

$$\theta_{(D_G,C_G)} \overset{+}{\leftarrow} -\Delta_{\theta_{(D_G,C_G)}}(\mathcal{L}_{D_G} + \mathcal{L}_{C_G} + \mathcal{L}_{\mathrm{CS}} + \mathcal{L}_{\mathrm{LGP}}) \tag{14}$$

$$\theta_G \overset{+}{\leftarrow} -\Delta_{\theta_G}(\mathcal{L}_G + \mathcal{L}_{C_G} + \mathcal{L}_{\mathrm{recon}} + \mathcal{L}_{\mathrm{CS}} + \mathcal{L}_{\mathrm{RS}} + \mathcal{L}_{\mathrm{CR}}) \tag{15}$$

$$\theta_{D_{\mathrm{CR}}} \overset{+}{\leftarrow} -\Delta_{\theta_{D_{\mathrm{CR}}}}\mathcal{L}_{\mathrm{CR}} \tag{16}$$

$\mathcal{L}_{\mathrm{KL}}$ represents the KL term of a variational autoencoder [24]; $\mathcal{L}_{\mathrm{KL}}$ is scaled down by a small hyperparameter such that it does not reduce the reconstruction ability [23]. $\mathcal{L}_M$ and $\mathcal{L}_{\mathrm{recon}}$ are provided in Sec. 2.1. In case of $\mathcal{L}_{\mathrm{recon}}$, $\hat{x}$ is from $G(C_C(x), E(x))$.

4 Experiments

We evaluated the performance of our method and the comparison methods on three datasets: MNIST [25] (digit number as a class label), SVHN [32] (digit number as a class label), and CelebA [29] (gender as a class label). In CelebA, we used gender as a class label. Unlike person identity or other CelebA attributes, such as smiling and beard, several factors should be considered when selecting whether an observation is a female or a male. To demonstrate the effectiveness of our study, a classification task is required, in which several consistent factors are considered to classify an observation.

Table 2. Classification accuracy of $p(y|r)$. We trained a classifier using r of ours and the comparison methods. The classifier would have a random chance for test datasets if there is no class-relevant information in r (lower is better).

Architecture	MNIST	SVHN	CelebA
Cycle-VAE [20]	0.176	0.436	0.793
ML-VAE [4]	0.717	0.445	0.786
LORD [12]	0.099	0.163	0.517
Ours	0.099	0.099	0.501
Random Chance	0.100	0.100	0.500

We predetermined the sizes of c and r for the three datasets. In MNIST and SVHN, $c \in \mathbb{R}^4$ and $r \in \mathbb{R}^8$. In CelebA, $c \in \mathbb{R}^8$ and $r \in \mathbb{R}^{16}$. Sec. S6 of the supplementary discusses the sizes of c and r in detail. In addition, Sec. S1 of the supplementary provides details of experiment information of our work.

4.1 Informativeness

We measured the classification performance of our model and compared it with those of ResNet-18 [16] and DeepCaps [36]. Our method shows similar accuracies on all three datasets, as shown in Table. 1. Our model can provide an explanation of the model's prediction without degradation in classification performance. We verified the informativeness of c using quantitative and qualitative methods.

Quantitative Experiments By following the protocol from Cycle-VAE [20], we trained a simple classifier to classify class labels from the residual representation r. This experiment was conducted to evaluate whether the class-relevant information is present in r. For comparison, we used recent class-supervised disentanglement methods [20,4,12]. The details of hyperparameters used for the comparison methods are provided in Sec. S9 of the supplementary.

As shown in Table. 2, only our method preserved a random chance in all the three datasets. This is because unlike the comparison methods, we did not agree with and implement the assumption that intra-class variation can be ignored. For datasets created for disentanglement learning, such as Cars3D [37], SmallNorb [26], KTH [39], etc., class-relevant intra-class variation is certainly small. However, for complex real datasets, class-relevant intra-class variation is typical (even large). Empirically, the classification accuracies of SVHN and CelebA in Table. 2 show that the completely class-irrelevant r cannot be created when assumed that intra-class variation can be ignored. For further analysis, we measured the mutual information between y and c, r of our model: $I(c; y)$ and $I(r; y)$. The $c0$ to $c7$ in Fig. 3 represent each element of the output vector of the class capsule. As shown in Fig. 3(a, b, c), all the elements of c of our method are strongly correlated to y, and all the elements of r are uncorrelated to y for the datasets.

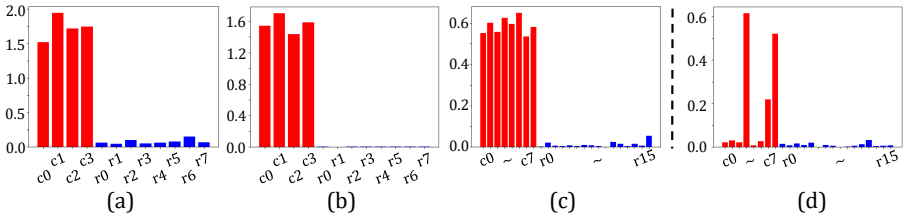

Fig. 3. Mutual information between y and c, r of (a) MNIST, (b) SVHN, (c) CelebA, and (d) CelebA w/o $\mathcal{L}_{\text{concept}}$ (higher is better for c; lower is better for r)

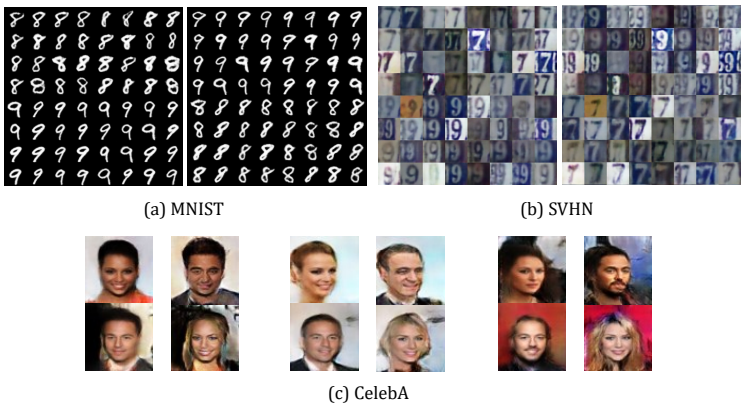

Fig. 4. Swapping. The images on the left of (a, b, c) are reconstructed using (c^i, r^i) and (c^j, r^j). The images on the right of (a, b, c) are generated by swapping c: (c^j, r^i) and (c^i, r^j).

As an ablation study, we tested the importance of $\mathcal{L}_{\text{concept}}$. As shown in Fig 3(d), some elements of c obtained without $\mathcal{L}_{\text{concept}}$ show low correlations to y. r still does not contain any class-relevant information, however, $\mathcal{L}_{\text{concept}}$ is required to encode class-relevant information to each element of c. Also, We tested the importance of $\mathcal{L}_M$ by replacing C_C with ResNet-18. The result is discussed in Sec. S4 of the supplementary.

Qualitative Experiments By swapping, we visually evaluated which factors of variation were encoded into c and r and whether they were semantically correct. From two test images of different class labels, we obtained (c_i, r_i) and (c_j, r_j), individually. Subsequently, we swapped c_i and c_j to generate new observations. The images on the left of Fig. 4(a, b, c) are generated using the original set: (c_i, r_i) and (c_j, r_j). The images on the right of Fig. 4(a, b, c) are generated by swapping: (c_j, r_i) and (c_i, r_j). It is clear that for MNIST, r contains factors of variation such as thickness and skew. For SVHN, r contains background color, font color, thickness, location, and skew. For CelebA, r contains background color and a person's face feature. We believe that the results show semantically

Fig. 5. Ablation study of D_{CR}. For (a), changes in rows overlap, whereas changes in rows are distinct in (b).

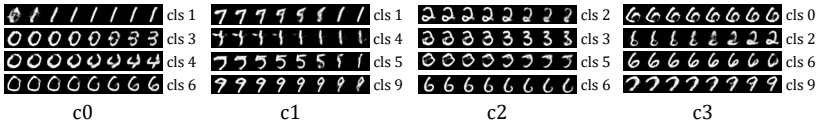

Fig. 6. Concepts learned for MNIST. c0: being a big circle, c1: being a straight line, c2: creating a small circle in the bottom part and c3: changing the upper part of the digit to a line.

correct disentanglement. A detailed analysis of c will be presented in Sec. 4.3. In addition, the T-SNE [30] results of r and c of our method are shown in Sec. S7. The measured FID [17] scores of the images by reconstruction, swapping, and random generation are provided in Sec. S8 of the supplementary.

4.2 Distinctness

Distinctness means that each concept should be represented by a single variable. The variable type varies in each method, which can be a prototype or a latent feature [6,31]. In our method, the single concept is represented by a single element of the class capsule. To enforce it, we used an additional discriminator, D_{CR}. Fig. 5(a) shows the images generated by G trained without D_{CR}. Each row of image (a) indicates each element of the class capsule, and the eight images of the row are the results of linear interpolation between -1 and 1. In Fig. 5(b), the change in each element is distinct from each other, whereas in Fig. 5(a), the elements overlap. For example, the change in the left-half of the first row and the change in the right-half of the fourth row in MNIST of Fig. 5(a) are the almost same. In such a case, the values for a single concept would be contradicting. This shows that D_{CR} is required to enforce distinctness between the elements. The same trend was observed in CelebA. The qualitative and quantitative results of CelebA are given in Fig. S2 and Table. S3 of the supplementary. In addition, the importance of all the six components is discussed in Sec. S2 of the supplementary.

4.3 Explainability

If explanations are provided based on concepts, these concepts should be human-understandable. In our setting, we demonstrate the learned concepts by linearly interpolating or analyzing data points of similar values. In Fig. 6, the concept

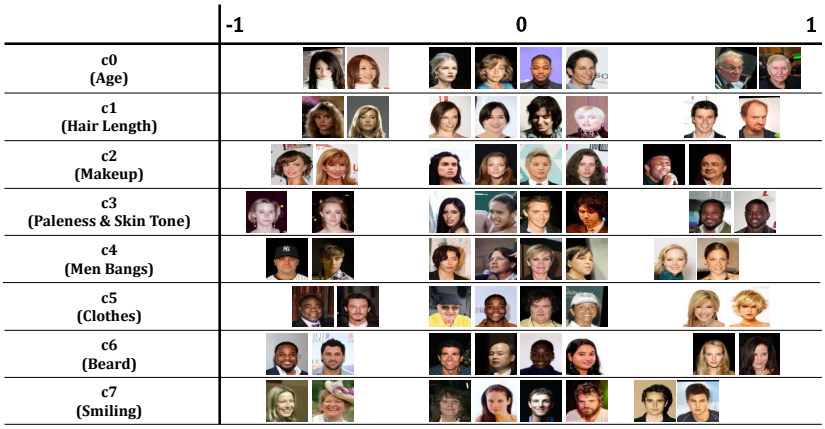

Fig. 7. Concepts learned for CelebA. For more details, $c2$ represents makeup. As the value approaches -1, people with heavy makeup appear. $c6$ represents a beard. As the value approaches -1, people who have a beard appear. Other concepts can be understood by verifying which image has a negative or positive value for each element.

of each element is shown by linear interpolation. In MNIST, r contains concepts such as thickness and skew, as described in Sec. 4.1. For c, the first element represents being a large circle. As the value approaches to -1, a majority of the digits are changed to zero. The second element represents being a straight line, and the third and fourth elements represent creating a small circle in the bottom part and changing the upper part of the digit to a line, respectively. By analyzing these results, we recognized that the factors relevant to MNIST classification are the size, number, and location of the circles and lines, and this finding also fits to SVHN in a very similar way (Sec. S5).

For CelebA, we show the concept of each element by analyzing a set of test images of certain values. We discovered the concepts such as age, hair length, makeup, paleness, skin tone, men hairstyle, clothes, beard, and smile, as shown in Fig. 7. Case $c1$ represents hair length. As the value approaches 1, a person with extremely short hair appears. Case $c4$ represents men's hairstyle. A person who has men bangs and perm typically exhibits a value less than 0. In case $c5$, the majority of men exhibit values less than 0. $c5$ is close to -1 when the person wears a suit and close to 1 when the person wears open-shoulder clothes.

We discovered an interesting phenomenon: In case $c0$, the element represents age. For persons appearing young, they typically have a value less than 0. Statistically, the number of females with values less than 0 was high, and the number of females with values greater than 0 was low. For males, the situation was vice versa. The model learned data bias from the CelebA dataset. When we analyzed the attribute named "young" of the CelebA dataset, we discovered an imbalance in the number of data, i.e., a ratio 2:1 (female:male). Similar to this case, we discovered an imbalance in CelebA attributes "pale skin" (3:1) and "smiling" (2:1), and these biases were encoded as class-relevant concepts ($c0$, $c3$, and $c7$).

		Age	Hair Length	Make up	Skin Tone	Men Bangs	Clothes	Beard	Smiling	Confidence
(a)	 F	-0.073	-0.271	-0.364	-0.469	0.306	0.175	0.408	-0.523	0.999
(b)	 M	0.439	0.411	0.325	0.300	-0.344	-0.370	-0.410	0.124	0.999
(c)	 F	-0.101	-0.261	-0.101	0.006	-0.027	0.073	-0.116	-0.063	0.335
(d)	 M	0.043	-0.016	-0.001	0.017	-0.054	0.040	0.001	0.024	0.087

Fig. 8. Explanations generated by iCaps for four samples. In the cases of (a, b), iCaps is accurate in the predictions. On the contrary, in the cases of (c, d), iCaps misclassified with a low confidence score. By analyzing the values of each concept of the samples, we can understand the high and low confidence of iCaps in making the predictions.

We discovered that our model can be used as a detector of hidden data bias; this will be investigated in future studies.

Herein, we demonstrate the explainability of our method using samples. In Fig. 8, we present classification success cases of a female and a male, as well as misclassification cases of a female and a male. By analyzing the values of the concepts, we understood why the model classified Fig 8(a) as a female with such high confidence. Fig 8(a) was predicted as a female owing to observations of long hair, pale skin, no men bangs, no beard, and a smiling face. In the misclassification case (d), the model showed very low confidence in the classification because the model could not find a strong relevance to any concepts.

5 Conclusion and Future Work

We propose a novel disentanglement method that the class-relevant subspace contains both class-relevant inter- and intra-class variation. Using the proposed method, we build a new interpretable model that provides explanations of the model's prediction based on class-relevant distinct concepts.

In addition, the generator of our model can generate an image of the desired combination of the concepts. Hence, it can be used for data augmentation or additional explanations. Also, we will keep analyzing the possibility of our model as a detector of data bias. In future studies, we try to improve reconstruction ability and further, add a sentence generation phase at the end so that the model can generate an explanation as a sentence automatically.

Acknowledgments. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (Ministry of Science and ICT) [2018R1A2B3001628], the Brain Korea 21 Plus Project in 2020, Samsung Advanced Institute of Technology and Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2019-0-01367, BabyMind), and AIR Lab in Hyundai Motor Company through HMC-SNU AI Consortium Fund.

References

1. Adebayo, J., Gilmer, J., Goodfellow, I., Kim, B.: Local explanation methods for deep neural networks lack sensitivity to parameter values. arXiv preprint arXiv:1810.03307 (2018)
2. Ahmed, K., Torresani, L.: Star-caps: Capsule networks with straight-through attentive routing. In: Advances in Neural Information Processing Systems. pp. 9098–9107 (2019)
3. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015)
4. Bouchacourt, D., Tomioka, R., Nowozin, S.: Multi-level variational autoencoder: Learning disentangled representations from grouped observations. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
5. Chang, C.H., Creager, E., Goldenberg, A., Duvenaud, D.: Explaining image classifiers by counterfactual generation (2018)
6. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. In: Advances in Neural Information Processing Systems. pp. 8928–8939 (2019)
7. Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., Abbeel, P.: Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In: Advances in neural information processing systems. pp. 2172–2180 (2016)
8. Dabkowski, P., Gal, Y.: Real time image saliency for black box classifiers. In: Advances in Neural Information Processing Systems. pp. 6967–6976 (2017)
9. Denton, E.L., et al.: Unsupervised learning of disentangled representations from video. In: Advances in neural information processing systems. pp. 4414–4423 (2017)
10. Fong, R., Patrick, M., Vedaldi, A.: Understanding deep networks via extremal perturbations and smooth masks. arXiv preprint arXiv:1910.08485 (2019)
11. Fong, R.C., Vedaldi, A.: Interpretable explanations of black boxes by meaningful perturbation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3429–3437 (2017)
12. Gabbay, A., Hoshen, Y.: Demystifying inter-class disentanglement. In: International Conference on Learning Representations (2020), `https://openreview.net/forum?id=Hyl9xxHYPr`
13. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in neural information processing systems. pp. 2672–2680 (2014)
14. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C.: Improved training of wasserstein gans. In: Advances in neural information processing systems. pp. 5767–5777 (2017)
15. Hahn, T., Pyeon, M., Kim, G.: Self-routing capsule networks. In: Advances in Neural Information Processing Systems. pp. 7656–7665 (2019)
16. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
17. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: Advances in neural information processing systems. pp. 6626–6637 (2017)

18. Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., Lerchner, A.: beta-vae: Learning basic visual concepts with a constrained variational framework. Iclr **2**(5), 6 (2017)
19. Jeong, T., Lee, Y., Kim, H.: Ladder capsule network. In: International Conference on Machine Learning. pp. 3071–3079 (2019)
20. Jha, A.H., Anand, S., Singh, M., Veeravasarapu, V.: Disentangling factors of variation with cycle-consistent variational auto-encoders. In: European Conference on Computer Vision. pp. 829–845. Springer (2018)
21. Kim, B., Gilmer, J., Wattenberg, M., Viégas, F.: Tcav: Relative concept importance testing with linear concept activation vectors (2018)
22. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In: International Conference on Machine Learning. pp. 2668–2677 (2018)
23. Kim, H., Mnih, A.: Disentangling by factorising. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 2649–2658. PMLR, Stockholmsmässan, Stockholm Sweden (10–15 Jul 2018)
24. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
25. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
26. LeCun, Y., Huang, F.J., Bottou, L.: Learning methods for generic object recognition with invariance to pose and lighting. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004. vol. 2, pp. II–104. IEEE (2004)
27. Li, O., Liu, H., Chen, C., Rudin, C.: Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In: Thirty-Second AAAI Conference on Artificial Intelligence (2018)
28. Lin, Z., Thekumparampil, K.K., Fanti, G., Oh, S.: Infogan-cr: Disentangling generative adversarial networks with contrastive regularizers. arXiv preprint arXiv:1906.06034 (2019)
29. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (December 2015)
30. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
31. Melis, D.A., Jaakkola, T.: Towards robust interpretability with self-explaining neural networks. In: Advances in Neural Information Processing Systems. pp. 7775–7784 (2018)
32. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning (2011)
33. Odena, A., Olah, C., Shlens, J.: Conditional image synthesis with auxiliary classifier gans. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2642–2651. JMLR. org (2017)
34. Petsiuk, V., Das, A., Saenko, K.: Rise: Randomized input sampling for explanation of black-box models. arXiv preprint arXiv:1806.07421 (2018)
35. Petzka, H., Fischer, A., Lukovnikov, D.: On the regularization of wasserstein GANs. In: International Conference on Learning Representations (2018), `https://openreview.net/forum?id=B1hYRMbCW`

36. Rajasegaran, J., Jayasundara, V., Jayasekara, S., Jayasekara, H., Seneviratne, S., Rodrigo, R.: Deepcaps: Going deeper with capsule networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 10725–10733 (2019)
37. Reed, S.E., Zhang, Y., Zhang, Y., Lee, H.: Deep visual analogy-making. In: Advances in neural information processing systems. pp. 1252–1260 (2015)
38. Sabour, S., Frosst, N., Hinton, G.E.: Dynamic routing between capsules. In: Advances in neural information processing systems. pp. 3856–3866 (2017)
39. Schuldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local svm approach. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. vol. 3, pp. 32–36. IEEE (2004)
40. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 618–626 (2017)
41. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 3145–3153. JMLR. org (2017)
42. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034 (2013)
43. Smilkov, D., Thorat, N., Kim, B., Viégas, F., Wattenberg, M.: Smoothgrad: removing noise by adding noise. arXiv preprint arXiv:1706.03825 (2017)
44. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806 (2014)
45. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 3319–3328. JMLR. org (2017)
46. Szabó, A., Hu, Q., Portenier, T., Zwicker, M., Favaro, P.: Challenges in disentangling independent factors of variation. arXiv preprint arXiv:1711.02245 (2017)
47. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European conference on computer vision. pp. 818–833. Springer (2014)
48. Zintgraf, L.M., Cohen, T.S., Adel, T., Welling, M.: Visualizing deep neural network decisions: Prediction difference analysis. arXiv preprint arXiv:1702.04595 (2017)