Appendix for "CS-Isolate: Extracting Hard Confident Examples by Content and Style Isolation"

Yexiong Lin[1] **Yu Yao**[2,3] **Xiaolong Shi**[1]
Mingming Gong[4] **Xu Shen**[5] **Dong Xu**[6] **Tongliang Liu**[1*]
[1]The University of Sydney; [2]Mohamed bin Zayed University of Artificial Intelligence;
[3]Carnegie Mellon University; [4]The University of Melbourne;
[5]Alibaba DAMO Academy; [6]The University of Hong Kong.

A A Theoretical View of Content and Style Isolation When Learning with Noisy Labels

Noisy data generative process. Recall the data generative process in our main paper, to learn latent factors by leveraging generative models. The generative model has to model the data generative process of noisy data. Firstly, we introduce the generative process of noisy data. We denote observed variables with gray color and latent variables with white color. Specifically, the content factor Z_c is generated by the latent label Y. The different style domain U_s give rise to the different style factor Z_s. Subsequently, the image X is generated by the combined influence of the style factor Z_s and the content factor Z_c. Noisy labels $\tilde{Y}$ are then generated based on the image X. In general cases, Z_c and Z_s can also have statistical or causal dependencies. We follow existing work that assumes the content factors are invariant across different styles [6].

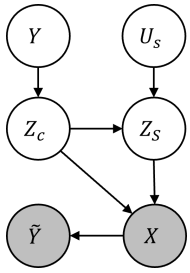


Figure 1: The noise data generative process.

Firstly, we introduce the concept of an *uncontrolled style factor*. This refers to a specific style factor, denoted $Z_{s'}$, that remains invariant when a data augmentation A is applied. To formalize this concept, consider an invertible function $f : \mathcal{Z} \times \mathcal{X}$. Let $\mathcal{A}$ denote a set of data augmentations, where each augmentation A is a subset ranging from 1 to M. Additionally, let $P(A)$ represent a probability distribution over the set of these augmentations $\mathcal{A}$. Now, consider $Z_{s'}$ as a subset of style factors drawn from a larger set Z_s. We partition the latent factors z of each instance x into three distinct parts: uncontrolled style factor $Z_{s'}$, content factor Z_c, and style factors influenced by data augmentation $Z_{s/s'}$, such that $f^{-1}(x) = [z_{s'}, z_c, z_{s/s'}] = z$. The term $z_{s/s'}$ denotes the set of style factors with style factors $z_{s'}$ excluded.

Definition 1 (Uncontrolled Style Factors)**.** *We say that a style factor $Z_{s'}$ as uncontrolled under the following conditions:*

For any augmentation $A \sim P(A)$, and for any instance x, the first $n_{s'}$ components of the inverse function $f^{-1}(x)$ remain unchanged even when $A(x)$ is applied, i.e., $f^{-1}(x)_{1:n_{s'}} = f^{-1}(A(x))_{1:n_{s'}}$.

Here, $f^{-1}(x)_{1:n_{s'}}$ is defined as the underlying partition that contains only and all the information related to the style factor $Z_{s'}$ of the instance x.

Why do confident examples encourage content-style isolation? Here, we explain the reason that confident examples encourage content-style isolation. Suppose that there exist some uncontrolled style factors that cannot be adjusted or manipulated through data augmentation. This implies that for an image x, these style factors remain unaffected, regardless of the data augmentation techniques

*Corresponding author: Tongliang Liu (tongliang.liu@sydney.edu.au).

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Algorithm 1 CS-Isolate-DM

Input: A noisy dataset $\tilde{\mathcal{S}}$, a style ID list U_s, a data augmentation set $\mathcal{A}$, confident threshold τ, total epoch T_{max} .

$f_{\psi_1}, f_{\psi_2}, q_{\phi_c^1}(Z_c|X), q_{\phi_c^2}(Z_c|X) \leftarrow \mathrm{WarmUP(S)}$;
For T = $1, \ldots, T_{max}$:
  $\mathcal{X}, \mathcal{U} \leftarrow \mathrm{Co-guessing}(\tilde{\mathcal{S}}, \mathrm{f}_{\psi_1}, \mathrm{f}_{\psi_2}, \mathrm{q}_{\phi_c^1}, \mathrm{q}_{\phi_c^2}, \tau)$;
  $\mathcal{X}', \mathcal{U}' \leftarrow \mathrm{MixUp}(\mathcal{X}, \mathcal{U})$;
  For k=1, 2:
    Sample $(x, \tilde{y}, u_c) \sim \tilde{\mathcal{S}}, u_s \sim U_s$;
    $\tilde{x} \leftarrow A_{u_s}(x)$;
    Feed $\tilde{x}$ to encoders $\hat{q}_{\phi_c^k}$ and $\hat{q}_{\phi_s^k}$ to content factors z_c and style factors z_s, respectively;
    Feed u_c and u_s to decoders $\hat{p}_{\theta_c^k}$ and $\hat{p}_{\theta_s^k}$ to get the prior $\hat{p}_{\theta_c^k}(Z_c|u_c)$ and $\hat{p}_{\theta_s^k}(Z_s|u_s)$;
    Feed z_c and z_s to decoders $\hat{p}_{\theta^k}$ to get the reconstruct image $\hat{\tilde{x}}$;
    Feed z_c to classifier head f_{ψ_k} to predicted label $\hat{y}$;
    Calculate the loss using Eq. 1 and update networks;

Output: The inference networks and classifier heads $q_{\phi_c^1}, q_{\phi_c^2}, f_{\psi_1}, f_{\psi_2}$.

employed. For instance, as discussed in our paper, in the CIFAR-10 dataset [3], some pictures with the label "horse" contain a person. In this context, the "person" acts as a style factor. Existing data augmentations cannot control this, as they cannot remove the person from images easily.

It's essential to understand that although data augmentation cannot control all style factors, it still offers the benefit of "partial isolation". If we consider a situation where the uncontrolled style factors are not effects of other style factors, then we can isolate other style factors affected by data augmentation from content factors after matching the data likelihood [5]. This result is a modified application of *block-identifiability* [6].

Specifically, let $\hat{Z}_{c,s'} := \hat{Z}_{1:n_c+n_{s'}}$ be a partition of learned representations of a generative model, where $\hat{Z}_{c,s'}$ contain *all* and *only* information about Z_c and $Z_{s'}$. Suppose that assumptions of Theorem 4.2 in [5] are satisfied, $\hat{Z}_{c,s'}$ is guaranteed to be learned, thereby allowing for the partial isolation of the remaining style factors, denoted as $Z_{s/s'}$.

Despite $\hat{Z}_{c,s'}$ is guaranteed to be learned, the information from the uncontrolled style factor $Z_{s'}$ is entangled with the content factor Z_c in the learned $\hat{Z}_{c,s'}$. To isolate this information further, the employment of *confident examples* is necessary.

Specifically, we can generate $\hat{Z}_{c,s'}$ using an invertible function according to the label $\hat{Y}$ of the confident example, and subsequently reconstruct image x. Following the data likelihood matching [5], the information related to the uncontrolled style factors becomes apparent due to the establishment of a one-to-one mapping between the label $\hat{Y}$ and $\hat{Z}_{c,s'}$. This consequently forces examples with identical labels to share the same $\hat{Z}_{c,s'}$, regardless of any alterations in the uncontrolled style factor Z'_s. This approach, therefore, ensures that styles changes don't affect the derived content representation for the same label.

It is worth mentioning that to fully isolate uncontrolled style factors and content factors, it requires that there exists confident examples with all possible uncontrolled style factors. This can be hard to achieve when learning with noisy labels. Therefore, in general, the selected confident examples can only encourage isolation but can not fully isolate uncontrolled style factors and content factors.

B Apply CS-Isolate to Existing Methods for Learning with Noisy Labels

Applying CS-Isolate to DivideMix. DivideMix [4] uses two classifiers to select confident examples for each other. To utilize the unlabeled data, they combine the semi-supervised technique MixMatch [1]. Specifically, the classifiers, after warmed up, are used to calculate the loss of examples. They use a Gaussian Mixture Model (GMM) to divide the examples into confident and unlabeled

Algorithm 2 CS-Isolate-Co

Input: A noisy dataset $\tilde{\mathcal{S}}$, a style ID list U_s, a data augmentation set $\mathcal{A}$, Total epoch T_{max} .

For T = $1, \ldots, T_{max}$:
  Fetch mini-batch $\bar{S}$ from $\tilde{\mathcal{S}}$;
  Obtain $\bar{S}_1 = \arg\min_{S':|S'|\geq R(T)|\bar{S}|} \ell(f_{\psi_1}, q_{\phi_c^1}, S')$;
  Obtain $\bar{S}_2 = \arg\min_{S':|S'|\geq R(T)|\bar{S}|} \ell(f_{\psi_2}, q_{\phi_c^2}, S')$;
  For k=1, 2:
    Sample $(x, \tilde{y}, u_c) \sim \bar{S}_k, u_s \sim U_s$;
    $\tilde{x} \leftarrow a_{u_s}(x)$;
    Feed $\tilde{x}$ to encoders $\hat{q}_{\phi_c^k}$ and $\hat{q}_{\phi_s^k}$ to content factors z_c and style factors z_s, respectively;
    Feed u_c and u_s to decoders $\hat{p}_{\theta_c^k}$ and $\hat{p}_{\theta_s^k}$ to get the prior $\hat{p}_{\theta_c^k}(Z_c|u_c)$ and $\hat{p}_{\theta_s^k}(Z_s|u_s)$;
    Feed z_c and z_s to decoders $\hat{p}_{\theta^k}$ to get the reconstruct image $\hat{\tilde{x}}$;
    Feed z_c and z_s to classification model f_{ψ_k} to predicted labels $\hat{y}$;
    Calculate the loss using Eq. 2 and update networks;

Output: The inference networks and classifier heads $q_{\phi_c^1}, q_{\phi_c^2}, f_{\psi_1}, f_{\psi_2}$.

Algorithm 3 CS-Isolate-Me

Input: A noisy training dataset $\tilde{\mathcal{S}}$, a noisy validation dataset $\tilde{\mathcal{S}}_v$, a style ID list U_s, a data augmentation set $\mathcal{A}$, iteration number T_{inner}, T_{outer} .

Initialize encoders ($\hat{q}_{\phi_c^0}$ and $\hat{q}_{\phi_s^0}$), decoders ($\hat{p}_{\theta_c^0}$, $\hat{p}_{\theta_s^0}$ and $\hat{p}_{\theta^0}$), a classification model f_{ψ_0} by using the noisy training data and early stopping;
For i = $1, \ldots, T_{outer}$:
  For j = $1, \ldots, T_{inner}$:
    Update the extracted confident examples $\mathcal{S}_l$ by using $\hat{q}_{\phi_c^{j-1}}$ and $f_{\psi_{j-1}}$;
    Train networks by using the loss in Eq. 3 on confident examples $\mathcal{S}_l$;
    Obtain $\hat{q}_{\phi_c^j}$ and f_{ψ_j} through the highest noisy validation accuracy throughout the training procedure;
    Break and output $\hat{q}_{\phi_c^{j-1}}$ and $f_{\psi_{j-1}}$ if the highest validation accuracy is non-increasing in the loop;
  Re-initialize encoders ($\hat{q}_{\phi_c^0}$ and $\hat{q}_{\phi_s^0}$), decoders ($\hat{p}_{\theta_c^0}$, $\hat{p}_{\theta_s^0}$ and $\hat{p}_{\theta^0}$), a classification model f_{ψ_0};
  Train networks by using the loss in Eq. 3 on confident examples $\mathcal{S}_l$;
  Obtain $\hat{q}_{\phi_c^0}$ and f_{ψ_0} through the highest noisy validation accuracy throughout the training procedure;
  Break and output $\hat{q}_{\phi_c^{j-1}}$ and $f_{\psi_{j-1}}$ if the highest validation accuracy is non-increasing in the loop;

Output: The inference network and the classification model q_{ϕ_c}, f_{ψ}.

examples. Finally, confident and unlabeled examples are used to train the models based on the MixMatch algorithm. Our method can be plugged into DivideMix easily. Specifically, we use a decoder $q_\phi(Z_c, Z_s|X)$ to obtain content factor Z_c and style factor Z_s. A classifier head f_ψ is used to predict labels, and only the content factors Z_c are used as input. The prior distribution of the content factors Z_c is conditional on the auxiliary variable U_c, *i.e.*, $P_{\theta_c}(Z_c|U_c)$, where U_c is the content ID. Similarly, the prior distribution of the style factor Z_s is conditional on the auxiliary variable U_s, *i.e.*, $P_{\theta_s}(Z_s|U_s)$, where U_s is the style ID. A decoder $P_\theta(X|Z_c, Z_s)$ is used to reconstruct input images. The combination of CS-Isolate and DivideMix is called CS-Isolate-DM. The loss function of CS-Isolate-DM is shown in Eq. 1. Algorithm 1 delineates the full algorithm.

$$\mathcal{L}_{dm} = \underbrace{\mathcal{L}_{\mathcal{S}_l} + \lambda_u \mathcal{L}_{\mathcal{S}_u} + \lambda_r \mathcal{L}_{\text{reg}}}_{\text{DivideMix loss}} + \lambda_{ELBO}\mathcal{L}_{ELBO} + \lambda_{ref}\mathcal{L}_{ref}. \tag{1}$$

Applying CS-Isolate to Co-Teaching. Co-Teaching [2] uses two classifiers to select confident examples for each other. Proposed CS-Isolate can be embedded in Co-Teaching easily. Similar to CS-Isolate-DM, we use a decoder $q_\phi(Z_c, Z_s|X)$ to obtain content factors Z_c and style factors Z_s. A

Table 1: Means and standard deviations (percentage) of classification accuracy on CIFAR-10.

	Sym-20%	Sym-40%	Sym-60%	Sym-80%	Pair-45%
CE	84.12 ± 0.20	78.82 ± 0.38	64.80 ± 0.47	47.39 ± 0.69	66.09 ± 0.96
Co-Teaching	88.74 ± 0.19	84.23 ± 0.81	77.93 ± 0.75	29.57 ± 1.39	76.65 ± 2.97
Forward	88.31 ± 0.23	82.73 ± 0.47	76.58 ± 1.59	47.18 ± 4.63	76.76 ± 4.94
T-Revision	87.83 ± 0.63	84.46 ± 0.73	77.39 ± 0.83	57.61 ± 3.93	72.81 ± 7.01
BLTM	76.54 ± 1.37	73.50 ± 1.38	58.94 ± 1.84	39.28 ± 4.00	67.97 ± 1.45
CausalNL	89.68 ± 0.09	86.37 ± 0.09	79.54 ± 0.09	29.72 ± 0.03	71.53 ± 0.37
Me-Momentum	91.44 ± 0.33	88.39 ± 0.34	82.58 ± 0.30	62.70 ± 0.59	66.41 ± 0.17
DivideMix	95.93 ± 0.04	94.51 ± 0.12	94.55 ± 0.07	92.43 ± 0.13	70.86 ± 0.87
CS-Isolate-DM	**96.05 ± 0.13**	**95.57 ± 0.12**	**94.65 ± 0.10**	**92.57 ± 0.10**	**87.54 ± 0.83**

Table 2: Precision ratio (percentage) of confident examples on CIFAR-10N.

	Worst	Aggregate	Random 1	Random 2	Random 3
Co-Teaching	89.49 ± 0.11	96.61 ± 0.04	96.06 ± 0.03	95.65 ± 0.02	96.01 ± 0.03
Me-Momentum	88.14 ± 1.10	97.16 ± 0.19	96.18 ± 0.47	95.96 ± 0.27	94.78 ± 0.47
CS-Isolate-Co	**90.88 ± 0.08**	**97.25 ± 0.04**	**96.95 ± 0.03**	**96.60 ± 0.06**	**96.82 ± 0.05**
CS-Isolate-Me	**90.74 ± 1.16**	**97.72 ± 0.07**	**96.64 ± 0.21**	**96.92 ± 0.17**	**96.14 ± 0.22**

Table 3: Recall ratio (percentage) of confident examples on CIFAR-10N.

	Worst	Aggregate	Random 1	Random 2	Random 3
Co-Teaching	89.11 ± 0.07	95.43 ± 0.04	92.59 ± 0.04	93.19 ± 0.02	93.00 ± 0.04
Me-Momentum	92.02 ± 2.63	96.90 ± 1.91	94.58 ± 2.07	95.80 ± 2.20	96.22 ± 1.82
CS-Isolate-Co	**90.53 ± 0.06**	**96.10 ± 0.02**	**93.44 ± 0.04**	**94.13 ± 0.06**	**93.79 ± 0.03**
CS-Isolate-Me	**95.76 ± 1.55**	**97.76 ± 0.79**	**97.10 ± 0.45**	**96.34 ± 0.81**	**96.70 ± 0.89**

classifier head f_ψ is used to predict labels, and only the content factors Z_c are used as input. The prior distribution of the content factors Z_c is conditional on the auxiliary variable U_c, *i.e.*, $P_{\theta_c}(Z_c|U_c)$, where U_c is the content ID. Similarly, the prior distribution of the style factor Z_s is conditional on the auxiliary variable U_s, *i.e.*, $P_{\theta_s}(Z_s|U_s)$, where U_s is the style ID. A decoder $P_\theta(X|Z_c, Z_s)$ is used to reconstruct input images. We call the combined method as CS-Isolate-Co. Let $\mathcal{S}_l$ be the confident examples selected by another classifier head. For each network, the loss is defined as:

$$\mathcal{L}_{co} = \underbrace{\mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathcal{S}}}[\mathbb{1}_{(x,\tilde{y})\in\mathcal{S}_l}\ell_{ce}(f_\psi \circ q_{\phi_c}(x), \tilde{y})]}_{\text{Co-Teaching loss}} + \lambda_{ELBO}\mathcal{L}_{ELBO}, \tag{2}$$

where ℓ_{ce} is the cross-entropy loss, $\mathbb{1}$ is the indicator function. The algorithm of CS-Isolate-Co-Teaching is summarized in Algorithm 2.

We use a PreAct ResNet-18 as the backbone. We used Adam with default parameters to optimize the encoder q_{ϕ_c}, classifier head f_ψ, the encoder q_{ϕ_s} and the decoder p_θ. The initial learning rate is 0.001, divided by 10 after 80 epochs.

Applying CS-Isolate to Me-Momentum. Me-Momentum proposes to use one classifier to select confident examples. Then the parameters of the classifier will be reinitialized, and the classifier will be trained on the confident examples. The confident examples and the parameters of the classifier are updated alternately. We combine the Me-Momentum with our method and call it CS-Isolate-Me. Let $\mathcal{S}_l$ be the confident examples selected by the classifier of the last iteration . For each network, the loss is defined as:

$$\mathcal{L}_{me} = \mathbb{E}_{(x,\tilde{y})\sim\tilde{\mathcal{S}}}[\mathbb{1}_{(x,\tilde{y})\in\mathcal{S}_l}\ell_{ce}(f_\psi \circ q_{\phi_c}(x), \tilde{y})] + \lambda_{ELBO}\mathcal{L}_{ELBO}. \tag{3}$$

where ℓ_{ce} is the cross-entropy loss, $\mathbb{1}$ is the indicator function. The algorithm of CS-Isolate-Me-Momentum is summarized in Algorithm 3.

Table 4: Means and standard deviations (percentage) of classification accuracy on CIFAR-10N.

	Worst	Aggregate	Random 1	Random 2	Random 3
Co-Teaching	82.04 ± 0.06	91.11 ± 0.10	89.61 ± 0.18	88.98 ± 0.11	89.49 ± 0.06
Me-Momentum	84.21 ± 0.70	91.34 ± 0.16	89.51 ± 0.42	90.14 ± 0.28	89.62 ± 0.31
CS-Isolate-Co	$\mathbf{83.93 \pm 0.17}$	$\mathbf{91.30 \pm 0.10}$	$\mathbf{90.57 \pm 0.14}$	$\mathbf{90.14 \pm 0.03}$	$\mathbf{90.76 \pm 0.15}$
CS-Isolate-Me	$\mathbf{86.95 \pm 0.13}$	$\mathbf{91.49 \pm 0.17}$	$\mathbf{90.30 \pm 0.12}$	$\mathbf{90.18 \pm 0.07}$	$\mathbf{90.22 \pm 0.21}$

Table 5: Data Augmentation Techniques

Data augmentation	**Description**
Shift scale rotation	Randomly shifts, scales, and rotates the image.
Random crop	Randomly crops the image to a specified height and width.
Horizontal flip	Horizontally flips the image randomly.
Random brightness contrast	Randomly changes the brightness and contrast of the image.
Color jitter	Randomly adjusts image color properties.
Random to gray	Converts the image to grayscale with a specified probability.

C Experiments

In this section, we first introduce the experiment results of the proposed methods, including the classification of CS-Isolate-DM on class-dependent label noise, sample selection quality of CS-Isolate-Co and CS-Isolate-Me on the real-world dataset CIFAR-10N, and classification performance of CS-Isolate-Co and CS-Isolate-Me on real-world the dataset CIFAR-10N. Second, we provide the details of data augmentation used in methods. Then, we conduct the ablation study for CS-Isolate-DM. Finally, we visualize the easy confident examples, content factors, and style factors.

C.1 Experiments on Class-dependent label noise

We report the classification performance of CS-Isolate-DM on class-dependent label noise, including symmetry-flipping noise and pair-flipping noise. The dataset used in the experiment is CIFAR-10. The experiment results are shown in Tab. 1. The experiment results demonstrate that CS-Isolate-DM can also perform well under class-dependent label noise.

C.2 Applying CS-Isolate to Existing Sample-Selection Methods

We combine our method with existing methods including DivideMix, Co-Teaching, and Me-Momentum.

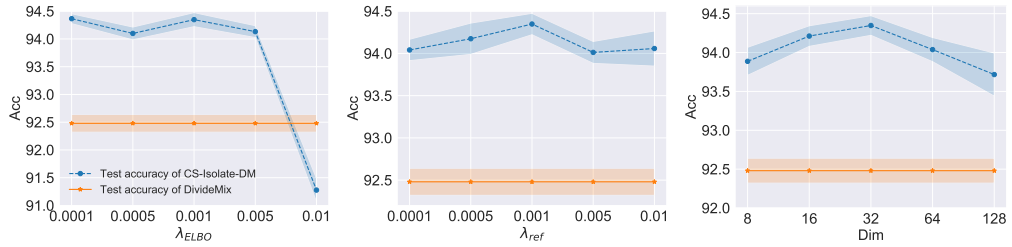
In the experiments for Co-Teaching and Me-Momentum, we use a PreAct ResNet-18 as the backbone. We use SGD with momentum 0.9 and weight decay 10^{-4} to optimize the encoder q_{ϕ_c} and classifier head f_ψ. We used Adam with default parameters to optimize the encoder q_{ϕ_s} and the decoder p_θ. The network is trained for 100 epochs. The initial learning rate for SGD is 0.01, and for Adam is 0.001. The learning rate is divided by 10 after 40 epochs and 80 epochs.

C.3 Improves Sample Selection Quality with CS-Isolate

We conducted experiments on CIFAR-10N, a dataset reflecting real-world label noise. We illustrate the precision and recall ratios of our confident examples in Tab. 2 and Tab. 3. By employing our method, existing methods achieve improvements in terms of precision and recall. The experiment results indicate that our approach can efficiently improve the quality and number of confident examples.

C.4 Comparison of Classification Performance

The test accuracy of the baseline methods, as well as the combination of our proposed methods and the baselines, is shown in Tab. 4. The results demonstrate that improving the quality of the confident examples by using our method boosts the classification performance of the existing methods.



(a) λ_{ELBO} (b) λ_{ref} (c) The dimensions of Z_c and Z_s

Figure 2: Ablation study on the hyper-parameters λ_{ELBO}, λ_{ref} and the dimensions of Z_c and Z_s.

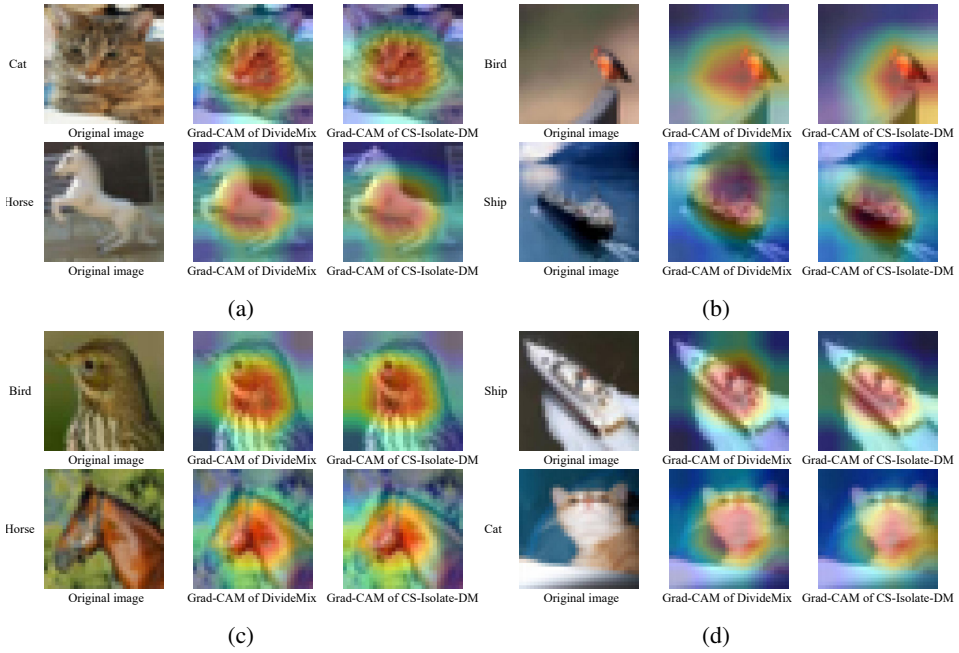


Figure 3: Grad-CAM visualizations of easy confident examples. Both CS-Isoalte-DM and DivideMix successfully identify these confident examples. The activation map of CS-Isolate-DM predominantly highlights semantic objects.

C.5 Data Augmentation Details

The detailed description of data augmentation techniques used in our method is shown in Tab 5. When generating a data augmentation $A_i \in \mathcal{A}$, the probabilities to apply shift scale rotation, random crop and horizontal flip, random brightness contrast, color jitter, and random to gray are 0.5, 1, 0.5, 0.5, 0.8, and 0.2, respectively, then the implementation details of the data augmentations will be recorded for the replaying. For instance, if a data augmentation A_i flips the image horizontally, its behavior will be recorded. When the data augmentation A_i is used during the training process, the images used to train the network will be applied a horizontal flip.

C.6 Ablation Study

In this subsection, we present the results of the ablation study on the hyper-parameters λ_{ELBO}, λ_{ref} and the dimensions of Z_c and Z_s. The experiments are conducted on the real-world dataset CIFAR-10N with the noise type "worst". The experiment results are shown in Fig. 2. The experiment results show that λ_{ELBO} and λ_{ref} are not sensitive in the range from 0.0005 to 0.005. In our experiments, we set the value of both λ_{ELBO} and λ_{ref} as 0.001, which is the middle value between 0.0005 and 0.005. For the ablation study on the dimension of Z_c and Z_s, the test accuracy increases

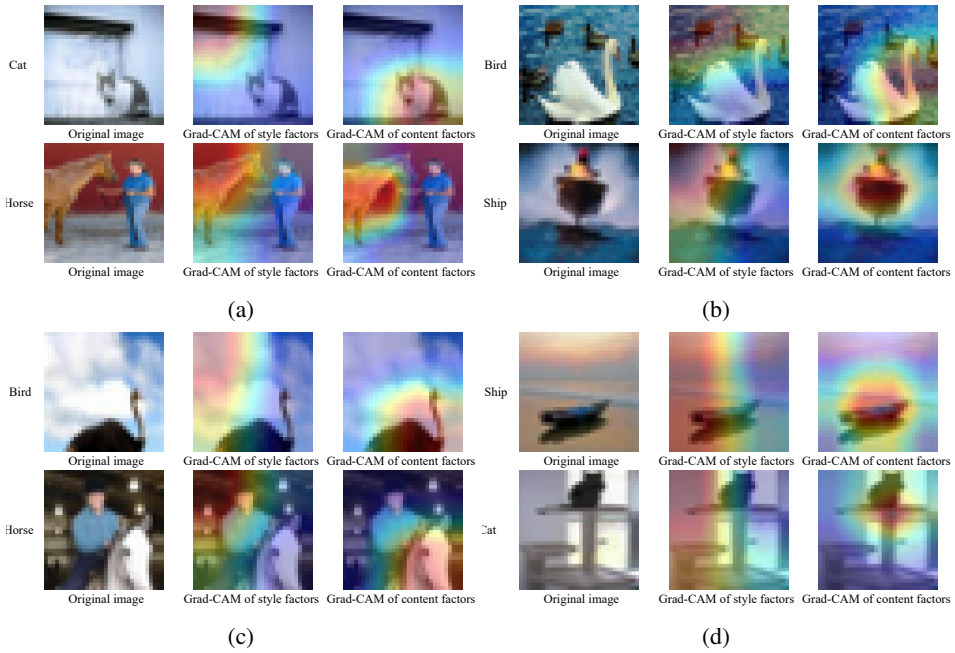


Figure 4: Grad-CAM Visualizations of Style and Content Factors for CS-Isolate-DM. The activation map corresponding to the content factor prominently highlights semantic objects, indicating the model's emphasis on capturing meaningful context. Conversely, the activation maps associated with the style factor predominantly focus on non-object pixels across the images. The results show that CS-Isolate-DM can isolate the content factors from styles.

gradually until the dimension is 32. After the dimension is larger than 64, the test accuracy decreases. We set the dimension as 32 in our experiments.

C.7 Visualization of Grad-CAM on Easy Confident Examples

We visualize the Grad-CAM for CS-Isolate-DM and DivideMix on easy confident examples. The easy confident examples are the examples identified successfully by both CS-Isolate-DM and DivideMix. The dataset is CIFAR-10N, and the noisy type is "worst". Fig. 3 shows the visualizations of easy confident examples. When the confident example is easy, the Grad-CAM visualizations for CS-Isolate-DM and DivideMix do not differ significantly. The activation maps for both CS-Isolate-DM and DivideMix can focus on the objects. However, when the confident example is hard, only the activation maps for CS-Isolate-DM can focus on the objects, which has already been shown in the main paper.

C.8 Visualization of Grad-CAM for Content Factors and Style Factors

We visualize the Grad-CAM of Style and Content Factors for CS-Isolate-DM. The visualization results are shown in Fig. 4. Grad-CAM of content factors mainly concentrates on the objects, while Grad-CAM of style factors mainly concentrates on other pixels in the images. The visualization results demonstrate that CS-Isolate-DM can isolate content and style factors successfully.

References

[1] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. *arXiv preprint arXiv:1905.02249*, 2019.

[2] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *NeurIPS*, pages 8527–8537, 2018.

[3] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[4] Junnan Li, Richard Socher, and Steven CH Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *ICLR*, 2019.

[5] Francesco Locatello, Ben Poole, Gunnar Rätsch, Bernhard Schölkopf, Olivier Bachem, and Michael Tschannen. Weakly-supervised disentanglement without compromises. In *International Conference on Machine Learning*, pages 6348–6359. PMLR, 2020.

[6] Julius Von Kügelgen, Yash Sharma, Luigi Gresele, Wieland Brendel, Bernhard Schölkopf, Michel Besserve, and Francesco Locatello. Self-supervised learning with data augmentations provably isolates content from style. *Advances in neural information processing systems*, 34:16451–16467, 2021.