

539 Appendix

540 A. Contribution Summary

541 Our contributions are summarized as follows:

- 542 • We observe some weaknesses in existing DA methods and address them by proposing a novel
543 method, named RrED, which introduces the diffusion model into the BUDA setup and strengthens
544 the target model’s reasoning ability through our two-stage learning.
- 545 • Inspired by the improved human decision-making process, RrED is designed to consist of two
546 stages, namely DTR and SRM. DTR guides the target model’s learning process by rectifying diffusion
547 model reasoning errors and leveraging its knowledge. SRM corrects errors in the learning process
548 by leveraging the differential reasoning ability of the target model and samples generated by the
549 fine-tuned diffusion model.
- 550 • To evaluate the effectiveness of RrED, we conduct extensive experiments, achieving SOTA perfor-
551 mance on four benchmarks. Ablation studies further highlight the contributions of each component
552 and provide a detailed analysis of the relationship among them.

553 B. Supplement of Complete Experimental Results

554 As shown in Table 5, the comparison results demonstrate that our RrED effectively employs a
555 two-stage strategy guided by diffusion model for target model optimization, achieving significantly
556 greater improvements on the large-scale benchmark *VisDA-17* [54]. Furthermore, we observe that
557 RrED does not outperform some comparison methods [17, 19] on certain classes. We attribute this to
558 the fact that our target model is trained under the guidance of a diffusion model, which tends to focus
559 on broad class distinctions to enhance overall discriminative ability. In contrast, the distillation-based
560 method SEAL [17] exhibits slight overfitting to a few specific classes (e.g., the “bike” and “truck”
561 class), resulting in higher recognition accuracy on those classes but reduced performance on others.
562 The CLIP-based method AEM [19] demonstrates notable discriminative power on specific classes.
563 Based on our analysis of the CLIP model [20], we find that these classes are often overrepresented
564 during CLIP pretraining. For example, there is a strong similarity between the “person” class in the
565 target domain and pretraining classes such as “baseball player”, “bridegroom”, and “scuba diver”
566 in CLIP model. In contrast, classes that are absent (e.g., the “knife” class) or rarely seen (e.g., the
567 “plant” class) during pretraining tend to have much lower recognition accuracy. In addition, we

Table 5: The complete accuracies (%) on the *VisDA-17* using ResNet-101 backbone.

Method	Setting	P	D	plane	bike	bus	car	horse	knife	mcycle	person	plant	sktbrd	train	truck	Mean
Source-only	—	×	×	64.3	24.6	47.9	75.3	69.6	8.5	79.0	31.6	64.4	31.0	81.4	9.2	48.9
HMA [46]	U	×	×	97.6	88.4	84.3	76.0	98.4	97.1	91.3	81.4	97.0	96.7	88.8	60.7	88.1
DAPL [47]	U	✓	×	97.8	83.1	88.8	77.9	97.4	91.5	94.2	79.7	88.6	89.3	92.5	62.0	86.9
PDA [48]	U	✓	×	99.2	91.1	91.9	77.1	98.4	93.6	95.1	84.9	87.2	97.3	95.3	65.3	89.7
DATUM [49]	U	✓	✓	85.7	76.4	79.7	75.4	84.1	82.3	80.4	76.7	81.9	82.6	78.4	20.2	75.3
S-Fusion [26]	U	✓	✓	92.9	83.7	89.3	87.0	95.3	92.7	90.1	86.8	92.2	93.2	88.3	42.0	86.1
DACDM [24]	U	✓	✓	96.2	84.8	83.2	73.3	94.8	96.6	91.0	88.2	93.0	93.4	87.5	59.7	86.8
DAD [24]	U	✓	✓	97.4	89.6	92.2	91.6	97.3	97.0	95.1	89.8	97.2	96.9	93.7	42.5	90.0
PLUE [50]	SF	×	×	97.3	96.2	90.5	91.8	90.0	94.2	87.4	87.7	97.0	84.3	93.0	81.0	90.0
C-SFDA [10]	SF	×	×	97.6	88.8	86.1	72.2	97.2	94.4	92.1	84.7	93.0	90.7	93.1	63.5	87.8
SF(DA) ² [9]	SF	×	×	96.8	89.3	82.9	81.4	96.8	95.7	90.4	81.3	95.5	93.7	88.5	64.7	88.1
DIFO [51]	SF	✓	×	97.7	87.6	90.5	83.6	96.7	95.8	94.8	74.1	92.4	93.8	92.9	65.5	88.8
DINE [15]	BP	×	×	81.4	86.7	77.9	55.1	92.2	34.6	80.8	79.9	87.3	87.9	84.3	58.7	75.6
BETA [16]	BP	×	×	94.9	90.2	85.4	61.1	95.5	93.1	85.0	83.8	92.9	91.9	91.1	55.0	85.1
RFC [18]	BP	×	×	95.6	89.7	87.8	75.8	96.5	96.5	90.4	82.8	96.0	70.0	85.7	55.1	85.2
SEAL [17]	BP	×	×	97.9	92.2	88.0	73.5	97.1	96.1	92.4	85.7	93.9	95.6	91.2	66.4	89.2
AEM [19]	BP	✓	×	98.6	88.1	89.7	74.8	98.0	93.9	93.0	89.3	90.1	97.2	95.2	63.5	89.3
RrED	BP	✓	✓	97.5	91.9	88.1	88.0	98.1	96.9	94.3	88.8	96.6	96.6	94.1	63.8	91.2

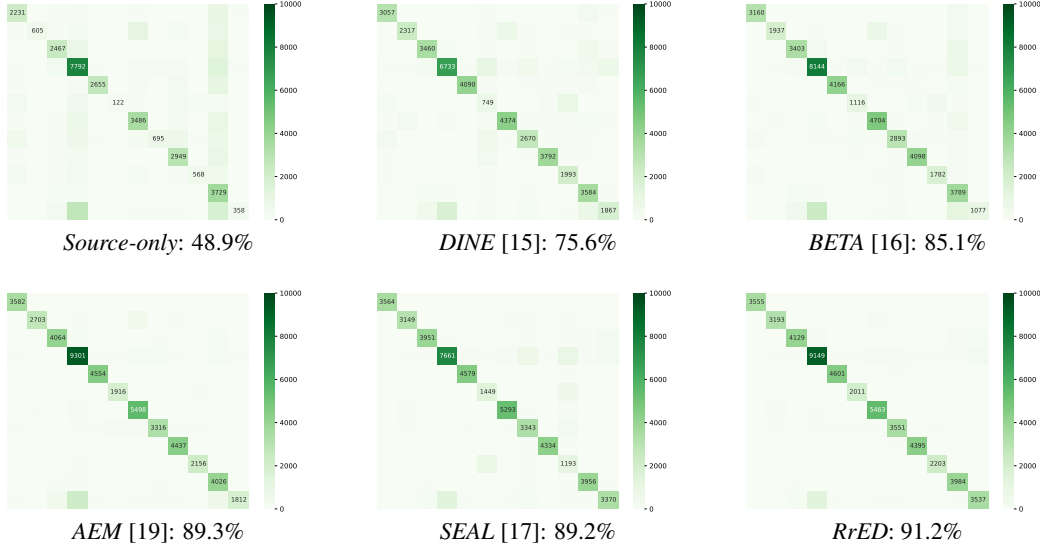


Figure 5: Classification results on *VisDA-17* are visualized with a confusion matrix. Note that all these results are obtained through the evaluation which is conducted in the same experimental environment. (Zooming in for a clear view)

568 supplement the classification visualization of the *VisDA-17* in Figure 5. For a fair comparison, all
 569 the classification visualizations are obtained in the same experimental environment. These results
 570 highlight that our RrED method substantially surpasses other BUDA approaches in improving class
 571 discrimination ability.

572 C. Theoretical Justifications

573 We provide theoretical justifications grounded in the generalization bound of reasoning to clarify the
 574 working mechanism of our algorithm.

575 First, we adopt PAC-Bayes theory [59] for the classification task to optimize the target model with
 576 the uncertainty estimation of the black-box predictor.

577 **Theorem 1** [60]. Given a target data distribution D_t , a hypothesis H , and a prior distribution π over
 578 the hypothesis space Θ . For any $\tau \in (0, 1]$ and $\lambda > 0$, with a probability at least $1 - \tau$ over the target
 579 samples $x_t \sim D_t$, for all posteriors ρ , we have:

$$\mathbb{E}_{\rho(H)} [\mathcal{L}(H)] \leq \mathbb{E}_{\rho(H)} [\tilde{\mathcal{L}}_{x_t}(H)] + \frac{1}{\lambda} [D_{KL}(\rho \parallel \pi) + \log \frac{1}{\tau} + \Psi_{x_t, \pi}(\lambda, n)], \quad (16)$$

580 where $\Psi_{x_t, \pi}(\lambda, n) = \log \mathbb{E}_{\pi(H)} \mathbb{E}_{x_t \sim D_t} [\exp(\lambda(\mathcal{L}(H) - \tilde{\mathcal{L}}(H)))]$.

581 **Lemma 1** [59]. The PAC-Bayes bound, involving constants τ and n , as introduced in Theorem 1, is
 582 minimized by the Bayesian posterior $\rho(H)$, which represents the distribution over Θ .

583 **Proof.** The Donsker-Varadhan’s change of measure states that for any measurable function $\phi : \Theta \rightarrow$
 584 \mathbb{R} , we have:

$$\mathbb{E}_{\rho(H)} [\phi(H)] \leq D_{KL}(\rho \parallel \pi) + \log \mathbb{E}_{\pi(H)} [\exp(\phi(H))]. \quad (17)$$

585 Thus, with $\phi(H) := \lambda(\mathcal{L}(H) - \tilde{\mathcal{L}}(H, x_t))$ and $\forall \rho$ over hypothesis space Θ , we have:

$$\begin{aligned} \mathbb{E}_{\rho(H)} \left[\lambda \left(\mathcal{L}(H) - \tilde{\mathcal{L}}(H, x_t) \right) \right] &= \lambda \left(\mathbb{E}_{\rho(H)} [\mathcal{L}(H)] - \mathbb{E}_{\rho(H)} [\tilde{\mathcal{L}}(H, x_t)] \right) \\ &\leq D_{KL}(\rho \parallel \pi) + \log \mathbb{E}_{\pi(H)} \left[\exp \left(\lambda \left(\mathcal{L}(H) - \tilde{\mathcal{L}}(H, x_t) \right) \right) \right]. \end{aligned} \quad (18)$$

586 For the non-negative random variable $\zeta_{\pi}(x_t) := \mathbb{E}_{\pi(H)} [\exp(\lambda(\mathcal{L}(H) - \tilde{\mathcal{L}}(H, x_t)))]$, we apply
 587 Markov’s inequality on it, and have:

$$\mathbb{P} \left(\zeta \leq \frac{1}{\tau} \mathbb{E}_{x_t \sim D_t} [\zeta_{\pi}(x_t)] \right) \geq 1 - \tau. \quad (19)$$

588 This implies that with probability at least $1 - \tau$ over the choice of $x_t \sim D_t$, we have $\forall \rho$ over
 589 hypothesis space Θ :

$$\mathbb{P} \left(\mathbb{E}_{\rho(H)} [\mathcal{L}(H)] \leq \mathbb{E}_{\rho(H)} [\tilde{\mathcal{L}}_{x_t}(H)] + \frac{1}{\lambda [D_{KL}(\rho || \pi) + \log \frac{1}{\tau} + \Psi_{x_t, \pi}(\lambda, n)]} \right) \geq 1 - \tau, \quad (20)$$

590 where $\Psi_{x_t, \pi}(\lambda, n) = \log \mathbb{E}_{\pi(H)} \mathbb{E}_{x_t \sim D_t} [\exp(\lambda(\mathcal{L}(H) - \tilde{\mathcal{L}}(H)))]$, and we prove the statement of
 591 Theorem 1. During target model training, as just as in Eq. (2), we utilize \mathcal{M}_θ as the prediction of
 592 posterior distribution and $S(x_i)$ as the prediction of prior distribution. Therefore, the upper bound of
 593 our target model can be expressed as:

$$\frac{1}{N_t} \sum_{i=1}^{N_t} [\mathcal{L}_{other} + \frac{1}{\lambda} D_{KL}(\mathcal{M}_\theta(x_i) || S(x_i))], \quad (21)$$

594 where N_t is defined as the number of the target data x_t . Following previous BUDA works [40, 15–
 595 19], as presented in Eq. (2), λ is set to 1 in the BUDA task. Moreover, \mathcal{L}_{other} varies in different works.
 596 For example, in RrED, $\mathcal{L}_{other} = \mathcal{L}_{GC} + \mathcal{L}_{CC}$ in the first stage DTR, and $\mathcal{L}_{other} = \mathcal{L}_{GC} + \mathcal{L}_{SVI}$
 597 in the second stage SRM. In summary, this proof fills the theoretical knowledge gap regarding the
 598 black-box predictor in previous BUDA works.

599 **Generalization Bound.** Since our target model is trained in unlabeled target domain data and
 600 further generates fusion data with feature differences based on the diffusion and our FSG module, we
 601 denote $x_t \sim D_t$ as the real sample distribution of the target domain and $\tilde{x}_t \sim \tilde{D}_t$ as the generated
 602 fusion sample distribution of the target domain. And denote y_t as the predicted label of x_t . For
 603 the corresponding generated fusion samples, \tilde{y}_t are the predicted labels of the target domain. D_t
 604 is uploaded to a black-box predictor to obtain hard predictions from a source model trained on the
 605 source domain D_s , where $x_s \sim D_s$ as the sample distribution of the source domain. The pioneering
 606 study [61] on theoretical analysis for domain adaptation provide the generalization bound. Following
 607 [61], let H denote a hypothesis, which can be expressed as:

$$\epsilon_t(H, y_t) \leq \epsilon_s(H, y_s) + d_{n\Delta n}(D_t, D_s) + \varphi, \quad (22)$$

608 where φ denotes the shared error of the ideal joint hypothesis, $\varphi = \min(\epsilon_s(H, y_s), \epsilon_t(H, y_t))$.
 609 $d_{n\Delta n}(D_s, D_t) = 2 \sup_{H, H' \in \mathcal{H}} |\mathbb{E}_{x_s \sim D_s} [H(x_s) \neq H'(x_s)] - \mathbb{E}_{x_t \sim D_t} [H(x_t) \neq H'(x_t)]|$.
 610 $\epsilon_t(H, y_t)$ is the expected error of the target sample distribution; $\epsilon_s(H, y_s)$ is the expected error of
 611 the source sample distribution, which is obtained from the black-box predictor. In the BUDA setting,
 612 although we do not have the source domain data x_s , we can obtain hard predictions P_s from the
 613 black-box predictor. Therefore, according to the theory [62, 63] of source data absence, $\epsilon_s(H, y_s)$ is
 614 small and can be ignored, so we do not need to obtain x_s and y_s in BUDA.

615 Then, we model a generated fusion domain distribution \tilde{D}_t that is distributed similarly to the target
 616 distribution D_t . To reduce the classification error on the target domain, the distributions D_s , D_t , and
 617 \tilde{D}_t should be substantially similar to each other. Therefore, the generalization bound in RrED can be
 618 transformed into:

$$\epsilon_t(H, y_t) \leq \tilde{\epsilon}_t(H, \tilde{y}_t) + d_{n\Delta n}(D_t, \tilde{D}_t) + \varphi_1, \quad (23)$$

619 where $\varphi_1 = \min(\epsilon_t(H, y_t), \tilde{\epsilon}_t(H, \tilde{y}_t))$; $\tilde{\epsilon}_t(H, \tilde{y}_t)$ is the expected error of the generated fusion
 620 domain distribution, which can be expressed as:

$$\tilde{\epsilon}_t(H, \tilde{y}_t) \leq \epsilon_s(H, y_s) + d_{n\Delta n}(\tilde{D}_t, D_s) + \varphi_2, \quad (24)$$

621 where $\varphi_2 = \min(\tilde{\epsilon}_t(H, \tilde{y}_t), \epsilon_s(H, y_s))$. Thus, our final generalization bound can be defined as:

$$\epsilon_t(H, y_t) \leq \epsilon_s(H, y_s) + d_{n\Delta n}(D_t, \tilde{D}_t) + d_{n\Delta n}(\tilde{D}_t, D_s) + \varphi_1 + \varphi_2, \quad (25)$$

622 For Eq. (25), we analyze each component in detail in this paragraph:

623 • $\epsilon_s(H, y_s)$ is the expected error of the source sample distribution. During the training of the source
 624 model, the error between the source domain data and its true labels is minimized by cross-entropy
 625 loss. Thus, in the early stages of training, we can obtain good training results for the source samples
 626 through the black-box predictor. As the training progresses, the model gradually adapts to the
 627 distribution of the target domain with Adaptive Label Smoothing (ALS) [15]. The ALS maintains
 628 source domain knowledge, enabling the model to learn target domain knowledge while preventing

Algorithm 1 RrED for BUDA task.

Input: Target samples $D_t = \{(x_i)\}_{i=1}^{N_t}$; black-box hard predictions P_s ; diffusion model with the predictor p_θ ; multi-modal model $\mathcal{V}_\theta \in \{\text{image encoder } \mathcal{I}_\theta, \text{ text encoder } \mathcal{T}_\theta\}$; and target model $\mathcal{M}_\theta \in \{\text{feature extractor } f_\theta, \text{ prediction classifier } c_\theta\}$.

Parameter: Training epoch e ; learnable prompt text embedding L ; model parameter θ ; and hyperparameters γ, r .

```

1: Initialize: initialize the smooth label repository  $S$  with  $P_s$ ; initialize  $\mathcal{V}_\theta$  with  $L$  and  $S$ ; diffusion model
   initializes to generate data  $x_{i,(g)}$  corresponding to  $x_i$ ;
2: ===== Diffusion-Target model Rectification =====
3: for  $i \leftarrow 1$  to  $e/2$  do
4:   Get target sample  $x_i$  and the sample predictions  $y_i$  using  $\mathcal{M}_\theta$ :  $y_i = f_\theta(c_\theta(x_i))$ ;
5:   Get generated fusion sample  $\tilde{x}_i$  to fuse  $x_{i,(g)}$  and  $x_i$  using Eqs. (6)-(7);
6:   Update the smooth label repository  $S$  using Eq. (1);
7:   Get fusion sample predictions  $\tilde{y}_i$  using  $\mathcal{M}_\theta$ :  $\tilde{y}_i = f_\theta(c_\theta(\tilde{x}_i))$ ;
8:   Fine-tune  $\mathcal{V}_\theta$  by minimizing  $\mathcal{L}_{\mathcal{V}_\theta}$  with  $p_\theta(x_i)$  using Eqs. (8)-(9):  $\min_{\mathcal{I}_\theta} \max_{\mathcal{T}_\theta} \mathcal{L}_{\mathcal{V}_\theta}$ ;
9:   Optimize  $\mathcal{M}_\theta$  by minimizing  $\mathcal{L}_{\mathcal{M}_\theta(DTR)}$  with  $p_\theta(x_i)$  using Eq. (13):  $\min_{f_\theta} \max_{c_\theta} \mathcal{L}_{\mathcal{M}_\theta(DTR)}$ ;
10: end for
11: ===== Self-Rectifying Reasoning Model =====
12: Initialize: Replace the original text encoder in the diffusion model with the fine-tuned text encoder with
   prompt word embeddings;
13: for  $i \leftarrow e/2$  to  $e$  do
14:   Get target sample  $x_i$  and the sample predictions  $y_i$  using  $\mathcal{M}_\theta$ :  $y_i = f_\theta(c_\theta(x_i))$ ;
15:   Get generated fusion sample  $\tilde{x}_i$  to fuse  $x_{i,(g)}$  and  $x_i$  using Eqs. (6)-(7);
16:   Update the smooth label repository  $S$  using Eq. (1);
17:   Get fusion sample predictions  $\tilde{y}_i$  using  $\mathcal{M}_\theta$ :  $\tilde{y}_i = f_\theta(c_\theta(\tilde{x}_i))$ ;
18:   Assign different weights  $w$  according to the similarities between  $y_i$  and  $\tilde{y}_i$  using Eq. (14);
19:   Optimize  $\mathcal{M}_\theta$  by minimizing  $\mathcal{L}_{\mathcal{M}_\theta(SRM)}$  with  $w$  using Eq. (15):  $\min_{f_\theta} \max_{c_\theta} \mathcal{L}_{\mathcal{M}_\theta(SRM)}$ ;
20: end for
Output: Target model  $\mathcal{M}_\theta$ .

```

629 the forgetting of source domain knowledge. Therefore, according to theories [62, 63], $\epsilon_s(H, y_s)$ is
630 small in the whole training.

631 • Instead of reducing $d_{n\Delta n}(D_t, D_s)$ in Eq. (16), our goal is to reduce $d_{n\Delta n}(D_t, \tilde{D}_t)$ and
632 $d_{n\Delta n}(\tilde{D}_t, D_s)$. For $d_{n\Delta n}(D_t, \tilde{D}_t)$, it depends on the expected error of the disagreement be-
633 tween two hypothesis on the target data and the generated fusion data distribution of the target
634 domain. During the whole training, we design the FSG module to determine which regions
635 should be composed of synthetic images. \tilde{D}_t is generated from D_t , preserving key features of
636 D_t while adding differential features generated by the diffusion model. Therefore, the distribu-
637 tion divergence $d_{n\Delta n}(D_t, \tilde{D}_t)$ is small. For $d_{n\Delta n}(\tilde{D}_t, D_s)$, we can obtain that $d_{n\Delta n}(\tilde{D}_t, D_s) =$
638 $2 \sup_{H, H' \in \mathcal{H}} |\mathbb{E}_{\tilde{x}_t \sim \tilde{D}_t} [H(\tilde{x}_t) \neq H'(\tilde{x}_t)] - \mathbb{E}_{x_s \sim D_s} [H(x_s) \neq H'(x_s)]|$. As the training progresses,
639 DTR aligns x_t and \tilde{x}_t by continuously minimizing the cross-entropy loss to facilitate the target model's
640 training; SRM narrows the feature space distance between x_t and \tilde{x}_t by contrasting their differences,
641 while enhancing the model's discriminative and generalization abilities by increasing dissimilarities
642 with other samples. Therefore, $\mathbb{E}_{\tilde{x}_t \sim \tilde{D}_t} [H(\tilde{x}_t) \neq H'(\tilde{x}_t)] \approx \mathbb{E}_{x_t \sim D_t} [H(x_t) \neq H'(x_t)]$ and it is
643 continuously reduced during training by minimizing \mathcal{L}_{GC} and \mathcal{L}_{task} . Meanwhile, \mathcal{L}_{CC} and \mathcal{L}_{SVI}
644 prevent overfitting of the target model. For $\mathbb{E}_{x_s \sim D_s} [H(x_s) \neq H'(x_s)]$, according to the previous
645 works [15, 16], the ALS maintains a source knowledge base and use \mathcal{L}_{task} to maintain the balance
646 between source knowledge and target knowledge. Therefore, $\mathbb{E}_{x_s \sim D_s} [H(x_s) \neq H'(x_s)]$ always
647 maintains a small value during the whole adaptation phase.

648 • $\varphi_1 + \varphi_2$ denotes the shared error of the ideal joint hypothesis, which is assumed to be a sufficiently
649 small constant that reflects the complexity of the hypothesis space [62].

650 D. The Whole Training Process

651 Our pseudocode for the training process is shown in Algorithm 1. In addition, our experimental and
652 main code are available in the Supplementary Material.

Table 6: The complete quantitative results of ablation study on the *Office-31* and *VisDA-17*.

\mathcal{L}_{GC}	$\mathcal{L}_{\mathcal{M}_\theta}$		FSG	FT	Office-31						VisDA-17	
	\mathcal{L}_{CC}	\mathcal{L}_{SVI}			A→D	A→W	D→A	D→W	W→A	W→D	Mean	Mean
Source only					79.9	76.6	56.4	92.8	60.9	98.5	77.5	48.9
✓			✓		97.8	85.2	66.6	97.0	72.1	97.5	86.0	71.2
	✓		✓		85.5	94.9	79.3	99.1	83.5	99.8	90.4	79.3
✓	✓				76.2	83.2	67.6	94.1	69.2	95.6	81.0	59.6
✓	✓		✓		95.2	95.7	81.5	99.0	83.1	99.8	92.4	89.4
		✓	✓		92.7	88.5	67.7	97.9	74.5	99.6	86.9	67.8
✓		✓	✓		96.9	94.1	73.7	97.3	81.6	99.8	90.6	85.4
	✓	✓	✓		85.3	84.9	66.7	97.0	71.9	98.0	84.0	80.2
	✓	✓	✓	✓	87.3	84.0	65.6	96.3	74.2	98.6	84.3	81.7
✓	✓	✓	✓		96.8	94.1	82.5	99.1	84.1	99.8	92.7	88.7
✓	✓	✓		✓	73.1	85.7	69.7	95.3	65.2	97.9	81.2	76.8
✓	✓	✓	✓	✓	97.8	95.9	83.7	99.1	84.5	99.8	93.5	91.2

E. Supplement of Complete Quantitative Ablation Experimental Results

As shown in Table 6, we report the complete quantitative results of ablation, and all the results include the task-specific loss. FSG is the key module of our work to prevent the diffusion-generated images from causing irreversible negative effects. The effective knowledge learning of the target model through \mathcal{L}_{GC} and \mathcal{L}_{CC} can only be achieved when FSG is utilized. When \mathcal{L}_{CC} is not used, the combined effect of \mathcal{L}_{task} and \mathcal{L}_{GC} enforces rapid sample clustering, which leads to overfitting of the target model. \mathcal{L}_{CC} mitigates the sample enrichment effect to improve target model’s generalization. In this regard, both \mathcal{L}_{task} and \mathcal{L}_{GC} can benefit from this process. \mathcal{L}_{SVI} is to integrate interactive learning with the samples generated by the fine-tuned diffusion model. \mathcal{L}_{SVI} becomes effective only when combined with fine-tuning. Experimental results show that this combination yields significant performance gains on large-scale dataset *VisDA-17*, while improvements on small-scale dataset *Office-31* are relatively limited. In summary, each component of our RrED contributes effectively to performance improvement and is indispensable.

F. Implementation Details.

We implement our RrED based on PyTorch and conduct all experiments using an NVIDIA GeForce RTX4090 GPU. For fair comparison, the backbone network is initialized following the protocol in [15], employing the ImageNet [64] pre-trained ResNet architectures: ResNet-50 for *Office-31*, *Office-Home*, and *DomainNet*, and ResNet-101 for *VisDA-17*. The optimization configuration employs SGD with a momentum of 0.9, a weight decay of 1e-3, and differentiated learning rates, where the learning rate is set to 1e-4 for the feature extractor f_θ and 1e-3 for the classifier c_θ . Following [16, 17], we set the bottleneck dimension to 256, the batch size to 64, the static momentum coefficient μ to 0.6, and the number of warm-up epochs to 3. To facilitate our joint multi-modal model CLIP [20] for fine-tuning, we choose Stable Diffusion v-1.5 [22] as the diffusion model. The strength of the noise addition is set to 0.6 in the diffusion model. For the diffusion predictor [42] and the fine-tuned text encoder we introduced, we keep their parameters frozen during the whole training. During the fine-tuning process, we follow [51, 19] to set the number of context tokens m to 4. All the reported quantitative results are obtained by averaging multiple runs with random seeds [2023, 2024, 2025].

G. More Visual Comparisons and Further Analysis

As shown in Figure 6, we use t-SNE [65] technique to visualize the distribution of target samples in the feature space. Compared with previous methods, the discrimination ability of the target model for target samples with similar features has been significantly improved under the training of our RrED algorithm. Moreover, as can be clearly observed from the graph, due to the enhanced generalization ability of the model after being trained by RrED, the differences between different classes become more pronounced, and the distances between samples of the same class become more

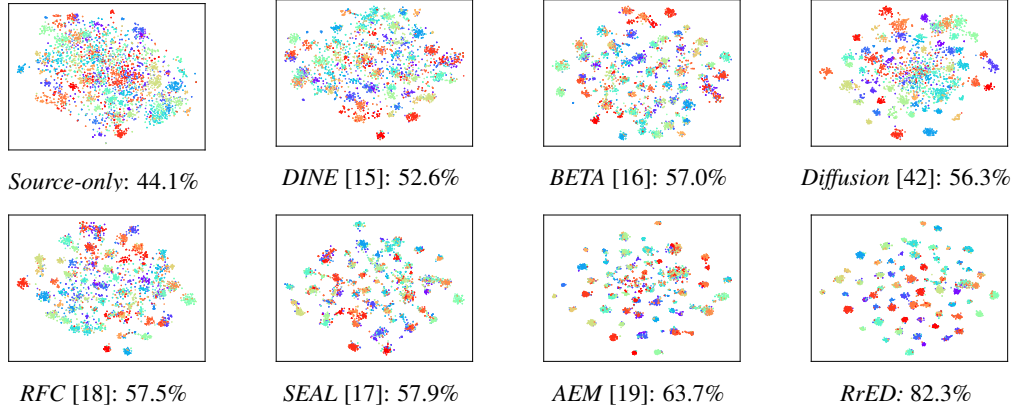


Figure 6: The feature visualization on the Office-Home (A→C) using the t-SNE [65]. Herein, the points represent target samples and the different colors correspond to their ground-truth classes. RrED introduces diffusion into BUDA and fine-tunes it, ultimately achieving remarkable improvement.

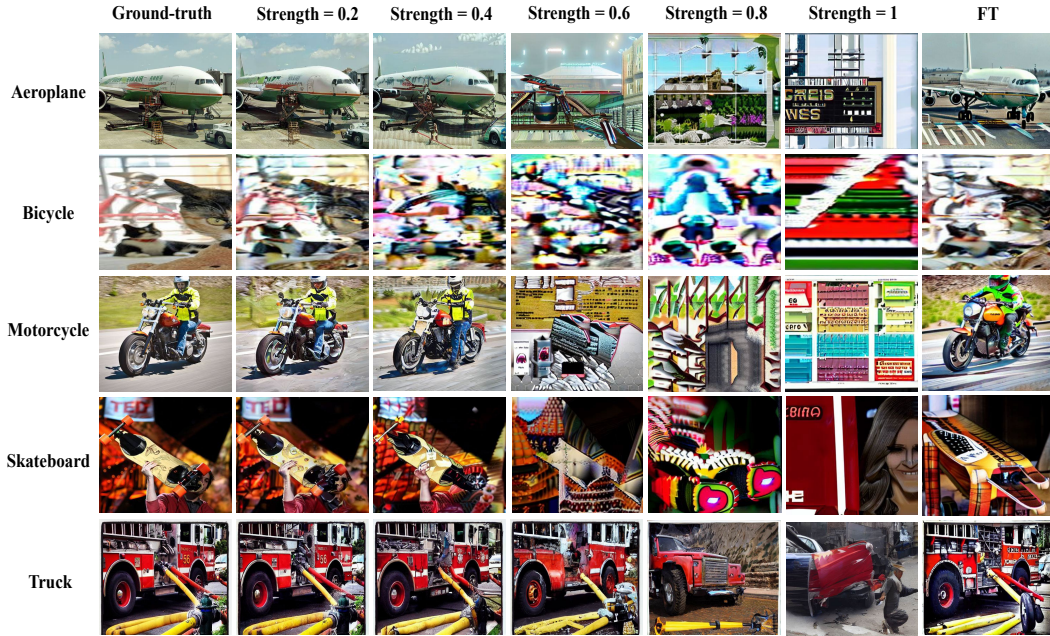


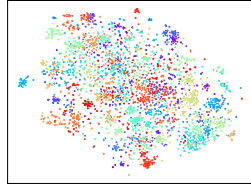
Figure 7: We present images generated by the diffusion model on the *VisDA-17* under varying noise strengths, along with those produced when the noise strength is 0.6 after our fine-tuning.

compact. Compared to the previous method [42] that directly applies diffusion model for prediction, our RrED exhibits superior model generalization and class discrimination capabilities. Therefore, we conclude that the target model trained by RrED achieves significant performance improvement in the high-security BUDA setting.

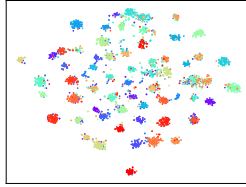
Next, we discuss our method’s exploration of the diffusion model to further demonstrate the superiority of our approach. As shown in Figure 7, we show the images that are generated by the diffusion model on the *VisDA-17* under varying noise strengths. When the noise level is too low, the images generated by the diffusion model are too similar to the target domain images, providing limited benefit for enhancing the model’s reasoning ability. When the noise level is too high, the images generated by the diffusion model differ drastically from those in the target domain and may even contain unrelated objects. Directly using such images can irreversibly disrupt the discriminative ability of the target model. *How can we effectively utilize the diffusion model to guide the target model in enhancing its reasoning ability while preventing its potential negative effects?* This is the problem our work

Table 7: Results of computational cost comparison on the *VisDA-17* with the ResNet-101 backbone. The batch size is 64.

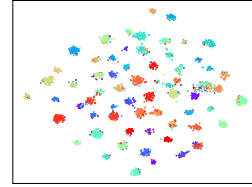
Method	Space (MiB)	Time (s/epoch)	Accuracy (%)
DINE	9881MiB	124s	75.6
BETA	20247MiB	1101s	85.1
SEAL	Over 24G	-	89.2
AEM	13747MiB	672s	89.3
RrED (Stage 1)	17654MiB	312s	89.4
RrED (Stage 2)	11721MiB	201s	91.2



Source-only: 44.1%



Stage 1: 79.8%



Stage 2: 82.3%

Figure 8: The feature distribution evolutions of different stages on the *Office-Home* (A→C) using t-SNE [65]. Herein, the points represent target samples and the different colors correspond to their ground-truth classes.

RrED aims to solve. For this, FSG serves as the key module to preclude the diffusion-generated images from bringing about irreversible adverse effects. FSG retains the regions of interest for the model, allowing the target model to maintain image discernibility even under higher noise levels in diffusion. Meanwhile, by fine-tuning the text encoder in the diffusion model, RrED enable it to better understand the content to be generated while maintaining its generative capabilities. As shown in Figure 7, the images generated by the fine-tuned diffusion model exhibit greater diversity, more distinct features, and fewer interfering objects. This allows the target model, in the second phase SRM, to first recognize the simpler generated images and then further distinguish the more challenging target images.

H. Computational Cost Comparison and Optimization Evolution

We supplement the computational cost comparison of the *VisDA-17* [54] in Table 7. For a fair comparison, all the results are obtained in the same experimental environment. In Table 7, we document the maximum GPU space usage, the average runtime cost, and the best accuracy of each comparison method. When adapting to the *VisDA-17* dataset, it is worth noting that the comparison methods have consumption-related limitations. BETA [16] operates in two computationally intensive stages: the first stage is the initialization, which requires initialization of the two models due to their mutually-distilled network structures; the second stage is the two-step process, which requires distillation and fine-tuning for each epoch. SEAL [17] is highly resource-intensive, and its official code cannot complete the adaptation task on *VisDA-17* under the same conditions with 24GB GPU memory. During the training of AEM [19], two classifiers are required: one classifier processes the output of the target model, while the other aligns with the predictions of the ViL model. Moreover, in each iteration, the weights of the overall model and the classifier weights need to be updated separately, resulting in consuming a significant amount of time. Compared with the previous BUDA methods, although RrED introduced the diffusion model in stage 1 to guide the learning of the target model, it still significantly reduced the time consumption by cutting out unnecessary calculation processes and optimizing loss functions. Moreover, in stage 2, after eliminating the resources consumed by fine-tuning and diffusion, RrED demonstrates extremely low overhead. These results demonstrate that our RrED significantly outperforms other BUDA methods in enhancing class discrimination ability at a relatively low cost.

729 In Figure 8, the optimization evolutions of feature distribution are presented. After the first stage
730 of training, the target model has learned the rich semantic knowledge in the diffusion model and
731 significantly improved its class discrimination ability. After the second stage of training, the scattered
732 data distribution boundaries stabilize around the nearest feature cluster centers, thus leading to the
733 samples with similar features exhibiting a more compact behavior. These results demonstrate the
734 superiority of the two-stage training in RrED and achieve the predefined objectives of each stage.

735 **I. Broader Impacts and Limitations**

736 Our work RrED focuses on the problem of Black-box Unsupervised Domain Adaptation (BUDA),
737 which provides better data privacy protection with more flexible portability compared with other
738 Domain Adaptation (DA) settings. Meanwhile, RrED demonstrates extremely superior performance,
739 significantly surpassing other DA methods. Inspired by research in neuroscience, RrED is specifically
740 designed for the classification task. While its effectiveness has been demonstrated through extensive
741 experiments and its theoretical soundness established, its applicability to other tasks remains an open
742 question. Therefore, we plan to further explore the practical utility of this algorithm in a broader
743 range of task scenarios.