Promising or Elusive? Unsupervised Object Segmentation from Real-world Single Images

Anonymous Author(s)
Affiliation
Address
email

Abstract

In this paper, we study the problem of unsupervised object segmentation from single images. We do not introduce a new algorithm, but systematically investigate the effectiveness of existing unsupervised models on challenging real-world images. We firstly introduce four complexity factors to quantitatively measure the distributions of object- and scene-level biases in appearance and geometry for datasets with human annotations. With the aid of these factors, we empirically find that, not surprisingly, existing unsupervised models catastrophically fail to segment generic objects in real-world images, although they can easily achieve excellent performance on numerous simple synthetic datasets, due to the vast gap in objectness biases between synthetic and real images. By conducting extensive experiments on multiple groups of ablated real-world datasets, we ultimately find that the key factors underlying the colossal failure of existing unsupervised models on real-world images is the challenging distributions of object- and scene-level biases in appearance and geometry. Because of this, the inductive biases introduced in existing unsupervised models can hardly capture the diverse object distributions. Our research results suggest that future work should exploit more explicit objectness biases in the network design.

1 Introduction

The capability of automatically identifying individual objects from complex visual observations is a central aspect of human intelligence [42]. It serves as the key building block for higher-level cognition tasks such as planning and reasoning [25]. In last years, a plethora of models have been proposed to segment objects from single static images in an unsupervised fashion: from the early AIR [19] and MONet [7] to the recent SPACE [34], SlotAtt [35], GENESIS-V2 [17], etc. They jointly learn to represent and segment multiple objects from a single image, without needing any human annotations in training. This process is often called perceptual grouping/binding or object-centric learning. These methods and their variants have achieved impressive segmentation results on numerous synthetic scene datasets such as dSprites [37] and CLEVR [29]. Such advances come with great expectations that the unsupervised techniques would likely close the gap with fully-supervised methods for real-world visual understanding. However, few work has systematically investigated the true potential of the emerging unsupervised object segmentation models on complex real-world images such as COCO dataset [33]. This naturally raises an essential question:

Is it promising or even possible to segment generic objects from real-world single images using (existing) unsupervised methods?

What is an object? To answer the above question involves another fundamental question: what is an object? Exactly 100 years ago in Gestalt psychology, Wertheimer [48] first introduced a set of Gestalt principles such as proximity, similarity and continuation to heuristically define visual

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Figure 1: The failure of SlotAtt [35] on three real-world images (right-hand side), although it can perfectly segment simple objects on three synthetic images (left-hand side).

data as objects. However, these factors are highly subjective, whilst the real-world generic objects are far more complex with extremely diverse appearances and shapes. Therefore, it is practically impossible to quantitatively define what is an object, i.e., the objectness, from visual inputs (e.g., a set of image pixels). Nevertheless, to thoroughly understand whether unsupervised methods can truly learn objectness akin to the psychological process of humans, it is vital to investigate the underlying factors that potentially facilitate or otherwise hinder the ability of unsupervised models. In this regard, by drawing on Gestalt principles, we instead define a series of new factors to quantitatively measure the complexity of objects and scenes in Section 2. By taking into account both the appearance and geometry of objects and scenes, our complexity factors explicitly assess the difficulty of segmenting a specific object. For example, a chair with colorful textures tend to have higher complexity than a single-color ball for unsupervised methods. With the aid of these factors, we extensively study whether and how existing unsupervised models can discover objects in Section 4.

What is the problem of unsupervised object segmentation from single images? A large number of models [50] aim to tackle the problem of unsupervised object segmentation from single images. They share several key problem settings: 1) all training images do not have any human annotations; 2) every single image has multiple objects; 3) each image is treated as a static data point without any dynamic or temporal information; 4) all models are trained from scratch without requiring any pretrained networks on additional datasets. Ultimately, the goal of these models is to segment all individual objects as accurate as the ground truth human annotations. In this paper, we regard these settings as the basic and necessary part of unsupervised object segmentation from single images, and empirically evaluate how successfully the existing models can exhibit on real-world images.

Our contributions and findings. This paper addresses the essential question regarding the potential of unsupervised segmentation of generic objects from real-world single images. Our key contributions and findings can be summarized as follows:

- We firstly introduce 4 complexity factors to quantitatively measure the difficulty of objects and scenes. These factors are key to investigate the true potential of existing unsupervised models.
- We extensively evaluate current unsupervised approaches in a large-scale experimental study. We implement 4 representative methods and train more than 130 models on 6 curated datasets from scratch. These datasets and pretrained models will be released for the community.
- We analyze our experimental results and find that:
  1) existing unsupervised object segmentation models cannot discover generic objects from single real-world images, although they can achieve outstanding performance on synthetic datasets, as qualitatively illustrated in Figure 1;
  2) the challenging distributions of both object- and scene-level biases in appearance and geometry from real-world images are the key factors incurring the failure of existing models;
  3) the inductive biases introduced in existing unsupervised models are fundamentally not matched with the objectness biases exhibited in real-world images, and therefore fail to discover the real objectness.

Scope of this research. This paper does not investigate unsupervised object discovery on saliency maps [47], static multi-views or dynamic videos [50]. Recent methods [9, 26] requiring pretrained models on monolithic object images such as ImageNet [39] are not evaluated as well.

2 Complexity Factors

As illustrated in the top row of Figure 2, an individual object, represented by a set of color pixels painted within a mask, can vary significantly given different types of appearance and geometric shape. A specific scene, represented by a set of objects placed within an image, can also differ vastly given different types of relative appearance and geometric layout between objects, as illustrated in the bottom row. Unarguably, such variation and complexity of appearance and geometry in both object level and scene level directly affects human's ability to precisely separate all objects. Naturally, the performance of unsupervised segmentation models are also expected to be influenced by the variation. In this regard, we carefully define the following two groups of factors to quantitatively describe the complexity of different datasets.

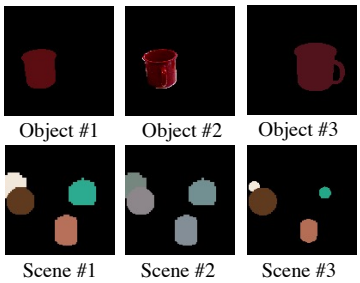


Figure 2: Complexity in appearance and geometry for objects and scenes.

2.1 Object-level Complexity Factors

As to a specific object, all its information can be described by appearance and geometry. Therefore we define the below two factors to measure the complexity of appearance and geometry respectively. Notably, both factors are nicely invariant to the object scale.

- **Object Color Gradient:** This factor aims to calculate how frequently the appearance changes within the object mask. In particular, given the RGB image and mask of an object, we firstly convert RGB into grayscale and then apply Sobel filter [41] to compute the gradients horizontally and vertically for each pixel within the mask. The final gradient value is obtained by averaging out all object pixels. Note that, the object boundary pixels are removed to avoid the interference of background. Numerically, the higher this factor is, the more complex texture and/or lighting effect the object has, and therefore it is likely harder to segment.
- **Object Shape Concavity:** This factor is designed to evaluate how irregular the object boundary is. Particularly, given an object (binary) mask, denoted as $\boldsymbol{M}_{obj} \in \mathbb{R}^{H \times W}$, we firstly find the smallest convex polygon mask ($\boldsymbol{M}_{cvx} \in \mathbb{R}^{H \times W}$) that surrounds the object mask using an existing algorithm [16], and then the object shape concavity value is computed as: $1 - \sum \boldsymbol{M}_{obj} / \sum \boldsymbol{M}_{cvx}$. Clearly, the higher this factor is, the more irregular object shape is, and segmentation is more tricky.

2.2 Scene-level Complexity Factors

As to a specific image, in addition to the object-level complexity, the spatial and appearance relationships between all objects can also incur extra difficulty for segmentation. We define the following two factors to quantify the complexity of relative appearance and geometry between objects in an image.

- **Inter-object Color Similarity:** This factor intends to assess the appearance similarity between all objects in the same image. Specifically, we firstly calculate the average color for each object, and then compute the pair-wise Euclidean distances of object colors, obtaining a $K \times K$ matrix where K represents the object number. The final inter-object color similarity is calculated by averaging the matrix excluding diagonal entries. Intuitively, the higher this factor is, the more similar all objects appear to be, the less distinctive each object is, and it is harder to separate each object.
- **Inter-object Shape Variation:** This factor aims to measure the relative geometry diversity between all objects in the image. We firstly calculate the diagonal length of each object bounding box, and then compute the pair-wise absolute differences for all object diagonal lengths, obtaining a $K \times K$ matrix. The final inter-object shape variation is the average of the matrix excluding diagonal entries. If this factor is higher, it means that objects within the image have more diverse and imbalanced sizes, and therefore segmenting both gigantic and tiny objects is likely more challenging.

By capturing the appearance and geometry in both object and scene levels, the four factors are designed to quantify the complexity of objects and images. In fact, they are chosen from more than 10 candidates because they are empirically more suitable to differentiate the gaps between synthetic

Table 1: Existing Unsupervised Models for Object Segmentation on Single Images. Each model includes different inductive biases, such as variational autoencoding (VAE), iterative inference (Iter), object relationship regularization (Rel), etc.

Factor-based Models		Inductive Biases			Layer-based Models		Inductive Biases		
		VAE	Iter	Rel			VAE	Iter	Rel
CST-VAE [27]	ICLRW'16	✓			Tagger [23]	NIPS'16		✓	
AIR [19]	NIPS'16	✓			RC [23]	ICLRW'16		✓	
SPAIR [13]	AAAI'19	✓			NEM [24]	NIPS'17		✓	
SuPAIR [43]	ICML'19	✓			MONet [7]	arXiv'19	✓		
GMIO [51]	ICML'19	✓	✓		IODINE [22]	ICML'19	✓	✓	
ASR [49]	NeurIPS'19	✓			ECON [45]	ICLRW'20	✓		✓
SPACE [34]	ICLR'20	✓			GENESIS [18]	ICLR'20	✓		✓
GNM [28]	NeurIPS'20	✓		✓	SlotAtt [35]	NeurIPS'20		✓	
SPLIT [10]	arXiv'20	✓			GENESIS-V2 [17]	NeurIPS'21	✓		✓
OCIC [2]	arXiv'20	✓		✓	R-MONet [40]	arXiv'21	✓		✓
GSGN [15]	ICLR'21	✓		✓	CAE [36]	arXiv'22			✓

and real-world scenes, and they eventually serve as key indicators to diagnose existing unsupervised models in Section 4. Calculation details of the four factors and other candidates are in appendix.

3 Experimental Design

3.1 Considered Methods

A range of works have explored unsupervised object segmentation in recent years. They are typically formulated as (variational) autoencoders (AE/VAE) [31] or generative adversarial networks (GAN) [21]. GAN based models [11, 3, 6, 44, 5, 46, 1] are usually limited to identifying a single foreground object and can hardly discover multiple objects due to the training instabilities, therefore not considered in this paper. As shown in Table 1, the majority of existing models are based on AE/VAE and can be generally divided into two groups according to the object representation:

- **Factor-based models**: Each object is represented by explicit factors such as size, position, appearance, etc, and the whole image is a spatial organization of multiple objects. Basically, such representation explicitly enforces objects to be bounded within particular regions.
- **Layer-based models**: Each object is represented by an image layer, i.e., a binary mask, and the whole image is a spatial mixture of multiple object layers. Intuitively, this representation does not have strict spatial constrains, and instead is more flexible to cluster similar pixels as objects.

In order to decompose the input images into objects, these approaches introduce different types of network architecture, loss functions, and regularization terms as inductive biases. These biases broadly include: 1) variational encoding which encourages the disentanglement of latent variables; 2) iterative inference which likely ends up with better scene representations over occlusions; 3) object relationship regularization such as depth estimation and autoregressive prior which aims at capturing the dependency of multiple objects; and many other biases. With different combinations of these biases, many methods have shown outstanding performance in synthetic datasets. Among them, we carefully select 4 representative models for our investigation: 1) AIR [19], 2) MONet [7], 3) IODINE [22], and 4) SlotAtt [35]. Implementation details are provided in appendix.

3.2 Considered Datasets

We consider two groups of datasets for extensive benchmarking and analysis: 1) three commonly-used synthetic datasets: dSprites [37], Tetris [30] and CLEVR [29], 2) three real-world datasets: YCB [8], ScanNet [14], and COCO [33], representing the small-scale, indoor- and outdoor-level real scenes respectively. Naturally, objects and scenes in different datasets tend to have very different types of biases. For example, the objects in dSprites tend to have the single-color bias, while COCO does not. Generally, the object-level biases can be divided as: 1) appearance biases including different textures and lighting effects, and 2) geometry biases including the object shape and occlusions. Similarly, the scene-level biases include: 1) appearance biases such as the color similarity between all objects, and 2) geometry biases such as the diversity of all object shapes. In fact, our complexity factors introduced in Section 2 are designed to well capture these biases. Table 2 qualitatively summarizes

the biases of selected datasets. We may hypothesize that the large gaps of biases between synthetic and real-world datasets would have a huge impact on the effectiveness of existing models.

Table 2: The object- and scene-level biases in appearance and geometry of the considered datasets.

		Synthetic Datasets			Real-world Datasets		
		dSprites [37]	Tetris [30]	CLEVR [29]	YCB [8]	ScanNet [14]	COCO [33]
Object-level Biases							
Appearance	Texture:	simple	simple	simple	diverse	simple	diverse
	Lighting:	no	no	synthetic	real	real	real
Geometry	Shape:	simple	simple	simple	simple	diverse	diverse
	Occlusion:	minor	no	minor	severe	severe	severe
Scene-level Biases							
Appearance	Similarity:	low	low	high	high	high	high
Geometry	Diversity:	low	low	low	high	high	high

To guarantee the fairness and consistency of all experiments, we carefully prepare all six datasets using the following same protocols. Preparation details for each dataset are provided in appendix.

- All images are rerendered or cropped with the same resolution of 128×128.
- Each image has about 2 to 6 solid objects with a blank background.
- Each dataset has about 10000 images for training, 2000 images for testing.

3.3 Considered Metrics

Having the six representative datasets and four existing unsupervised methods at hand, we choose the following metrics to evaluate the object segmentation performance: 1) AP score which is widely used for object detection and segmentation [20], 2) PQ score which is used to measure non-overlap panoptic segmentation [32], and 3) Precision and Recall scores. A predicted mask is considered correct if its IoU against a ground truth mask is above 0.5. All objects are treated as a single class. The blank background is not taken into account for fair comparison. To compute AP, we simply treat the mean value of the soft object mask as the object confidence score. Note that, the alternative metrics ARI [38] and segmentation covering (SC) [4] are not considered as they can be easily saturated.

4 Key Experimental Results

4.1 Can current unsupervised models succeed on real-world datasets?

First of all, we evaluate the selected four representative baselines on our six datasets separately. In particular, we train each model from scratch on each dataset separately. For fair evaluation, we carefully tune the hyperparameters of each model on every dataset and fully optimize the networks until convergence. Table 3 compares the quantitative results of all models. It can be seen that all methods demonstrate satisfactory object segmentation results on the three synthetic datasets, especially the recent strong baselines IODINE and SlotAtt. However, not surprisingly, all methods fail catastrophically on the three real-world datasets.

Table 3: Quantitative results of object segmentation from the four methods on six datasets.

	AIR [19]	MONet [7]	IODINE [22]	SlotAtt [35]
	AP / PQ / Pre / Rec	AP / PQ / Pre / Rec	AP / PQ / Pre / Rec	AP / PQ / Pre / Rec
dSprites	45.4 / 38.2 / 57.6 / 58.1	69.7 / 61.6 / 70.4 / 73.9	92.9 / 71.3 / 82.6 / 96.0	92.8 / 82.8 / 88.8 / 92.9
Tetris	25.2 / 23.4 / 36.8 / 39.9	85.9 / 75.8 / 85.1 / 89.7	52.2 / 37.9 / 48.0 / 61.8	94.3 / 79.9 / 90.5 / 94.4
CLEVR	46.4 / 44.3 / 67.4 / 52.5	39.0 / 37.3 / 65.6 / 42.8	82.8 / 73.0 / 77.5 / 87.4	91.7 / 82.9 / 90.8 / 92.7
YCB	0.0 / 0.6 / 1.1 / 0.8	3.1 / 7.0 / 9.8 / 1.2	1.8 / 3.9 / 6.2 / 7.3	9.2 / 13.5 / 20.0 / 26.2
ScanNet	2.7 / 6.3 / 15.6 / 7.3	24.8 / 24.6 / 31.0 / 40.7	10.1 / 13.7 / 18.6 / 24.4	5.7 / 9.0 / 12.4 / 18.3
COCO	2.7 / 6.7 / 14.3 / 8.6	11.8 / 12.5 / 16.1 / 21.9	4.0 / 6.3 / 9.9 / 10.8	0.8 / 3.5 / 5.3 / 7.3

Preliminary Diagnosis: In order to diagnose the colossal failure, we hypothesize that it is because of huge gaps in objectness biases between two types of datasets. In this regard, we quantitatively compute the distributions of our four complexity factors on the six datasets. In particular, the two

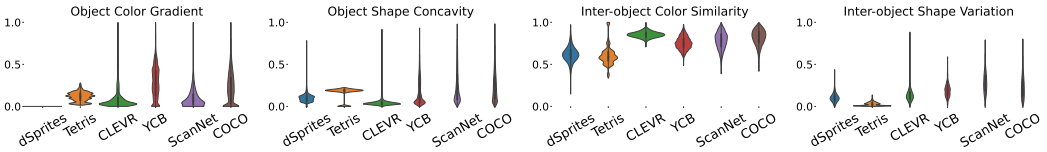


Figure 3: Distributions of the complexity factors on the three synthetic and real-world datasets.

object-level factors, *i.e.* Object Color Gradient and Object Shape Concavity, are computed for each object of the six training splits. The two scene-level factors, *i.e.* Inter-object Color Similarity and Inter-object Shape Variation, are computed for each image of the six training splits. From the distributions shown in Figure 3, we can see that, 1) for the two object-level factors, the three synthetic datasets tend to have extremely lower scores than the real-world datasets, which means that the synthetic objects are more likely have uniform colors and convex shapes; 2) for the two scene-level factors, the images in synthetic datasets tend to include less similar objects in terms of color, which means that multiple objects in real-world scenes are less distinctive in appearance. In addition, the multiple objects in synthetic scenes tend to have similar sizes, whereas real-world scenes usually have diverse object scales in single images. To validate whether these distribution biases are the true reasons incurring the failure, we conduct extensive experiments in Sections 4.2, 4.3 and 4.4.

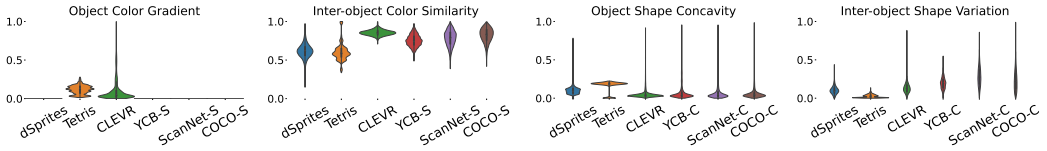
4.2 How do object-level factors affect current models?

In this section, we aim to verify to what extent the distributions of object-level factors affect the segmentation performance. In particular, we conduct the following three ablative experiments.

- *Ablation of Object Color Gradient*: For each object of the three real-world datasets, we only replace all pixel colors by its average color rgb value within each object mask, without touching the object shapes. In this way, the color gradients of each object are totally erased, thus removing the potential impact of Object Color Gradient. The three ablated datasets are: YCB-S / ScanNet-S / COCO-S.
- *Ablation of Object Shape Concavity*: For each object in the real-world datasets, we find the smallest convex hull [16] for its object mask and then fill the empty pixels by shifting the original object pixels. Basically, this ablation aims to only reduce the irregularity of object shapes, yet retaining the distributions of color gradients. The ablated datasets are: YCB-C / ScanNet-C / COCO-C.
- *Ablation of both Object Color Gradient and Shape Concavity*: We simply combine the above two ablation for each real-world object, getting datasets: YCB-S+C / ScanNet-S+C / COCO-S+C.

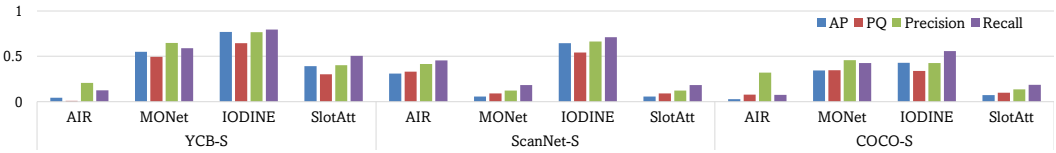
In Figure 4(a), we calculate the new distributions of both Object Color Gradient and Inter-object Color Similarity on the ablated datasets YCB-S/ScanNet-S/COCO-S. It can be seen that the object-level gradients become all zeros, even simpler than the three synthetic datasets. Yet, the distributions of Inter-object Color Similarity are almost the same as the original YCB/ScanNet/COCO. Similarly, Figure 4(b) shows that the distributions of Object Shape Concavity of ablated datasets now become similar to the synthetic datasets, while the distributions of Inter-object Shape Variation keep the same. Note that, for the ablation of both Object Color Gradient and Shape Concavity, the distributions will be the same as shown in Figure 4. Having the three groups of object-level ablated real-world datasets, we then evaluate segmentation performance of the four unsupervised models separately.

Analysis: As shown in Figure 5(a) and (c), we can see that once the pixels of real-world objects are replaced by its mean color, *i.e.* without any color gradients, the object segmentation performance has been significantly improved for almost all methods. From Figure 5(b), we can see that reducing the irregularity of real-world objects can also improve the object segmentation, although not significantly. Basically, these results show that exiting methods are more likely to learn the objectness represented by uniform colors and/or regular objects. However, comparing with Table 3, the results of current

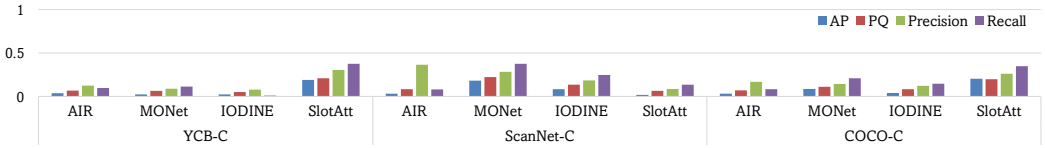


(a) Ablation of Object Color Gradient (b) Ablation of Object Shape Concavity

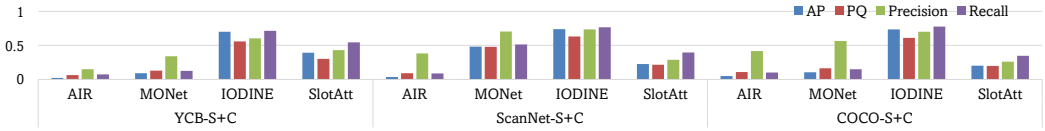
Figure 4: The new distributions of complexity factors after object-level ablations on real-world datasets.



(a) Quantitative results on the three datasets with Ablation of Object Color Gradient.



(b) Quantitative results on the three datasets with Ablation of Object Shape Concavity.



(c) Quantitative results on the three datasets with Ablation of both Object Color Gradient and Shape Concavity.

Figure 5: Results of four methods on the three groups of object-level ablated real-world datasets.

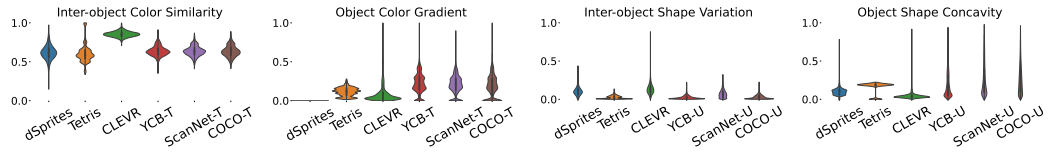
ablated datasets in Figure 5 still lag behind the synthetic datasets. This means that there must be some other factors that also potentially affect the object segmentation of existing models.

4.3 How do scene-level factors affect current models?

In this section, we turn to investigate to what extent the distributions of scene-level factors affect the segmentation performance. Particularly, we conduct the following three ablative experiments.

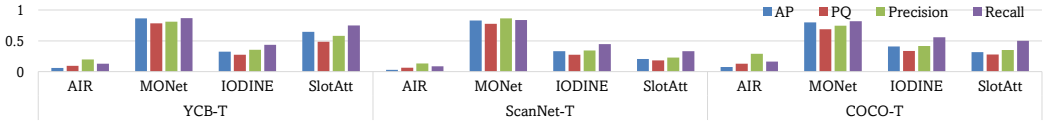
- *Ablation of Inter-object Color Similarity*: In each image of the three real-world datasets, we replace all object textures by a set of new distinctive textures from the existing DTD database [12]. In this way, the multiple objects look more distinctive in appearance, while the per-object texture gradients are roughly preserved. The ablated datasets are denoted: YCB-T / ScanNet-T / COCO-T.
- *Ablation of Inter-object Shape Variation*: In each image of the real-world datasets, we normalize the scales of multiple objects by shrinking or expanding the diagonal length of their bounding boxes, such that the new object sizes tend to be uniform. For each object, its shape and texture are linearly scaled up or down. Basically, this aims to remove the diversity of object sizes within single images. The ablated datasets are denoted as: YCB-U / ScanNet-U / COCO-U.
- *Ablation of both Inter-object Color Similarity and Shape Variation*: We simply combine the above two ablation strategies for each real-world image. Ablation details are in appendix. The ablated datasets are denoted as: YCB-T+U / ScanNet-T+U / COCO-T+U.

In Figure 6(a), we calculate the new distributions of both Inter-object Color Similarity and Object Color Gradient on the ablated datasets: YCB-T/ScanNet-T/COCO-T. It can be seen that the distributions of scene-level color similarity become nearly the same as the three synthetic datasets, while the distributions of object-level color gradients are still as high as the original real-world datasets YCB/ScanNet/COCO. Similarly, Figure 6(b) shows that the distributions of Inter-object Shape Variation of ablated datasets now become similar to the synthetic datasets, whereas the distributions of Object Shape Concavity are still the same as original real-world datasets YCB/ScanNet/COCO. Note

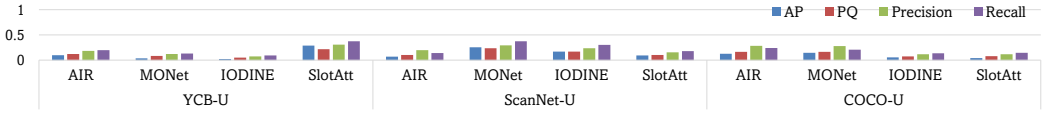


(a) Ablation of Inter-object Color Similarity (b) Ablation of Inter-object Shape Variation

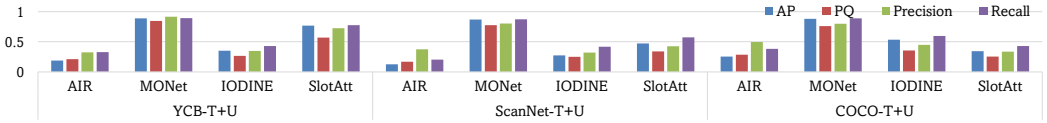
Figure 6: The new distributions of complexity factors after scene-level ablations on real-world datasets.



(a) Quantitative results on the three datasets with Ablation of Inter-object Color Similarity.



(b) Quantitative results on the three datasets with Ablation of Inter-object Shape Variation.



(c) Results on the three datasets with Ablation of both Inter-object Color Similarity and Shape Variation.

Figure 7: Results of four methods on the three groups of scene-level ablated real-world datasets.

that, for the ablation of both Inter-object Color Similarity and Shape Variation, the distributions will be the same as shown in Figure 6. Having these three groups of scene-level of ablated real-world datasets, we then separately evaluate the four unsupervised models.

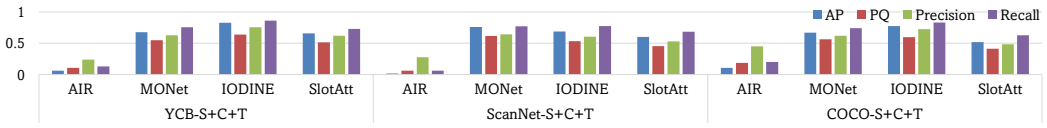
Analysis: As shown in Figure 7(a) and (c), we can see that once the textures of real-world images are replaced by more distinctive textures, *i.e.* with lower similarity between object appearance, the object segmentation performance has been surprisingly boosted remarkably almost for all methods. From Figure 7(b), we can see that normalizing object sizes over images can also reasonably improve the object segmentation performance. These results clearly show that existing unsupervised models significantly favor objectness with distinctive appearance in single images. However, compared with Table 3, the results on current scene-level ablated datasets are still inferior to the synthetic datasets, meaning that the scene-level factors alone are not enough to explain the performance gap.

4.4 How do object- and scene-level factors jointly affect current models?

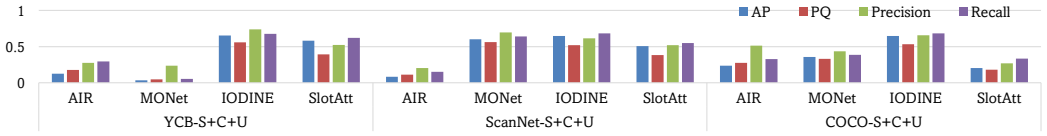
In this section, we aim to study how the object- and scene-level factors jointly affect the segmentation performance. In particular, we conduct the following three ablative experiments.

- *Ablation of Object Color Gradient, Object Shape Concavity and Inter-object Color Similarity*: In each image of the three real-world datasets, we replace the object color by averaging all pixels of the distinctive texture, and also replace the object shape with a simple convex hull. The ablated datasets are denoted: YCB-S+C+T / ScanNet-S+C+T / COCO-S+C+T.
- *Ablation of Object Color Gradient, Object Shape Concavity and Inter-object Shape Variation*: In each image of the three real-world datasets, we replace the object color by averaging its own texture, and modify the object shape as convex hull following by size normalization. The ablated datasets are denoted as: YCB-S+C+U / ScanNet-S+C+U / COCO-S+C+U.
- *Ablation of all four factors*: We aggressively combine all four ablation strategies and we get datasets: YCB-S+C+T+U / ScanNet-S+C+T+U / COCO-S+C+T+U.

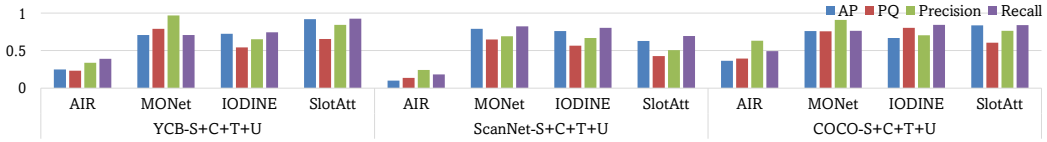
Analysis: Since these ablations are actually conducted independently, the new distributions of the four complexity factors on current jointly ablated datasets are exactly the same shown in Figures 4/6 respectively. Figure 8 shows the results of four unsupervised models on the three groups of object- and scene-level ablated datasets. From Figure 8 (a) and (b), we can see that combining the two object-level factors and either of scene-level factors to ablate the real-world datasets, the segmentation performance can be significantly improved as we expected. From Figure 8 (c), we can see that if the challenging real-world objects and images are ablated in both object- and scene-level, the segmentation performance of all unsupervised models achieves the same level with three synthetic datasets as shown in Table 3. Clearly, these three groups of experiments demonstrate that the colossal failure of unsupervised models on real-world images involves both object- and scene-level dataset biases. Existing unsupervised models cannot learn the challenging objectness in real-world images.



(a) Results for the Ablation of Object Color Gradient, Object Shape Concavity and Inter-object Color Similarity.



(b) Results for the Ablation of Object Color Gradient, Object Shape Concavity and Inter-object Shape Variation.



(c) Results for the Ablation of all object- and scene-level factors.

Figure 8: Results of four methods on the three groups of joint object- and scene-level ablated datasets.

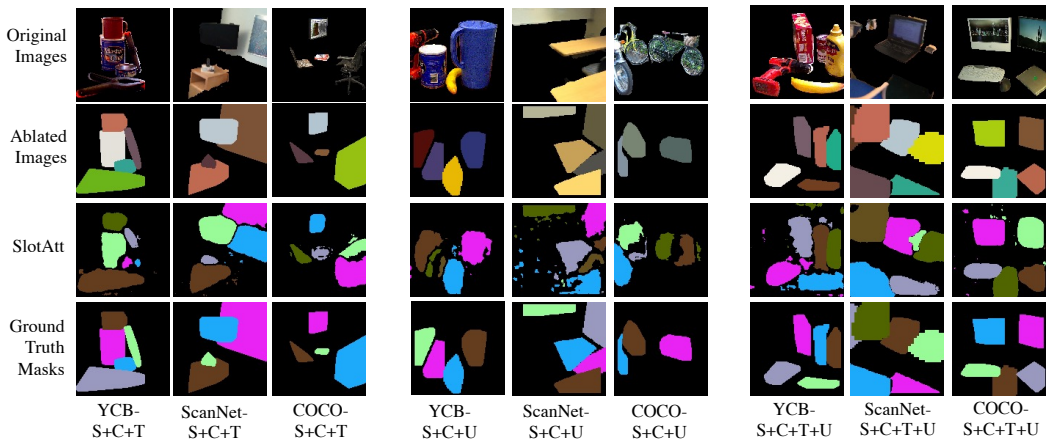


Figure 9: Results of SlotAtt on ablated real-world datasets with both object- and scene-level factors.

5 Conclusions

In this paper, we systematically show that the existing unsupervised methods are practically impossible to segment generic objects from single real-world images, and investigate the underlying factors that incur the catastrophic failure. In particular, we perform a large-scale experimental study with four representative unsupervised methods, four evaluation metrics on six datasets. With the aid of our carefully designed four object- and scene-level complexity factors, we conduct extensive experiments on multiple groups of ablated real-world objects and images, and safely conclude that the challenging distributions of both the object- and scene-level biases in appearance and geometry of real-world datasets are the key factors that current unsupervised models fail to segment real objects. Once these factors are removed from the real-world datasets, existing models can perform as excellent as on the synthetic datasets, as qualitatively illustrated in Figure 9. Fundamentally, this means that the inductive biases designed in existing unsupervised models are far from able to match with and capture the true objectness biases exhibited in real-world images. From this, we urge the community to exploit more explicit and reliable objectness biases in future unsupervised models, such as object motions or pretrained features from single-object-dominant datasets.

Broader Impact: This paper does not introduce a new algorithm, but extensively investigates the limit of existing unsupervised object segmentation models on real-world images. One limitation of this work is the lack of study about foreground-background segmentation. However, we believe that our work can still have a positive impact for researchers to design more practically useful models. A potential negative impact could be using our findings to attack an existing model.

References

[1] R. Abdal, P. Zhu, N. Mitra, and P. Wonka. Labels4Free: Unsupervised Segmentation using StyleGAN. *ICCV*, 2021.

[2] T. Anciukevicius, C. H. Lampert, and P. Henderson. Object-Centric Image Generation with Factored Depths, Locations, and Appearances. *arXiv:2004.00642*, 2020.

[3] R. Arandjelovic and A. Zisserman. Object Discovery with a Copy-Pasting GAN. *arXiv:1905.11369*, 2019.

[4] P. Arbeláez, M. Maire, C. Fowlkes, and J. Malik. Contour Detection and Hierarchical Image Segmentation. *TPAMI*, 33(5):898–916, 2011.

[5] S. Azadi, D. Pathak, S. Ebrahimi, and T. Darrell. Compositional GAN: Learning Image-Conditional Binary Composition. *IJCV*, 2020.

[6] A. Bielski and P. Favaro. Emergence of object segmentation in perturbed generative models. *NeurIPS*, 2019.

[7] C. P. Burgess, L. Matthey, N. Watters, R. Kabra, I. Higgins, M. Botvinick, and A. Lerchner. MONet: Unsupervised Scene Decomposition and Representation. 2019.

[8] B. Calli, A. Singh, J. Bruce, A. Walsman, K. Konolige, S. Srinivasa, P. Abbeel, and A. M. Dollar. Yale-CMU-Berkeley dataset for robotic manipulation research. *IJRR*, 36(3):261 – 268, 2017.

[9] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging Properties in Self-Supervised Vision Transformers. *ICCV*, 2021.

[10] R. Charakorn, Y. Thawornwattana, S. Itthipuripat, N. Pawlowski, and N. Dilokthanakul. An Explicit Local and Global Representation Disentanglement Framework with Applications in Deep Clustering and Unsupervised Object Detection. *arXiv:2001.08957*, 2020.

[11] M. Chen, T. Artières, and L. Denoyer. Unsupervised object segmentation by redrawing. *NeurIPS*, 2019.

[12] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, and A. Vedaldi. Describing Textures in the Wild. *CVPR*, 2014.

[13] E. Crawford and J. Pineau. Spatially Invariant Unsupervised Object Detection with Convolutional Neural Networks. *AAAI*, 2019.

[14] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner. ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes. *CVPR*, 2017.

[15] F. Deng, Z. Zhi, D. Lee, and S. Ahn. Generative Scene Graph Networks. *ICLR*, 2021.

[16] S. Eddins. Binary image convex hull. *https://blogs.mathworks.com/steve/2011/10/04/binary-image-convex-hull-algorithm-notes/*, 2011.

[17] M. Engelcke, O. P. Jones, and I. Posner. GENESIS-V2: Inferring Unordered Object Representations without Iterative Refinement. *NeurIPS*, 2021.

[18] M. Engelcke, A. R. Kosiorek, O. P. Jones, and I. Posner. GENESIS: Generative Scene Inference and Sampling with Object-Centric Latent Representations. *ICLR*, 2020.

[19] S. M. A. Eslami, N. Heess, T. Weber, Y. Tassa, K. Kavukcuoglu, and G. E. Hinton. Attend, Infer, Repeat: Fast Scene Understanding with Generative Models. *NIPS*, 2016.

[20] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge: A Retrospective. *IJCV*, 111:98–136, 2015.

[21] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Nets. *NIPS*, 2014.

[22] K. Greff, R. L. Kaufman, R. Kabra, N. Watters, C. Burgess, D. Zoran, L. Matthey, M. Botvinick, and A. Lerchner. Multi-object representation learning with iterative variational inference. *ICML*, 2019.

[23] K. Greff, A. Rasmus, M. Berglund, T. H. Hao, J. Schmidhuber, and H. Valpola. Tagger: Deep unsupervised perceptual grouping. *NIPS*, 2016.

[24] K. Greff, S. Van Steenkiste, and J. Schmidhuber. Neural Expectation Maximization. *NIPS*, 2017.

[25] K. Greff, S. van Steenkiste, and J. Schmidhuber. On the binding problem in artificial neural networks. *arXiv:2012.05208*, 2020.

[26] O. J. Hénaff, S. Koppula, E. Shelhamer, D. Zoran, A. Jaegle, A. Zisserman, J. Carreira, and R. Arandjelović. Object discovery and representation networks. *arXiv:2203.08777*, 2022.

[27] J. Huang and K. Murphy. Efficient inference in occlusion-aware generative models of images. *ICLR Workshops*, 2016.

[28] J. Jiang and S. Ahn. Generative neurosymbolic machines. *NeurIPS*, 2020.

[29] J. Johnson, B. Hariharan, L. van der Maaten, L. Fei-Fei, C. L. Zitnick, and R. Girshick. CLEVR: A Diagnostic Dataset for Compositional Language and Elementary Visual Reasoning. *CVPR*, 2017.

[30] R. Kabra, C. Burgess, L. Matthey, R. L. Kaufman, K. Greff, M. Reynolds, and A. Lerchner. Multi-Object Datasets. *https://github.com/deepmind/multi-object-datasets/*, 2019.

[31] D. P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *ICLR*, 2014.

[32] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic Segmentation. *CVPR*, 2019.

[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. *ECCV*, 2014.

[34] Z. Lin, Y.-F. Wu, S. V. Peri, W. Sun, G. Singh, F. Deng, J. Jiang, and S. Ahn. SPACE: Unsupervised Object-Oriented Scene Representation via Spatial Attention and Decomposition. *ICLR*, 2020.

[35] F. Locatello, D. Weissenborn, T. Unterthiner, A. Mahendran, G. Heigold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf. Object-Centric Learning with Slot Attention. 2020.

[36] S. Löwe, P. Lippe, M. Rudolph, and M. Welling. Complex-Valued Autoencoders for Object Discovery. *arXiv:2204.02075*, 2022.

[37] L. Matthey, I. Higgins, D. Hassabis, and A. Lerchner. dSprites - Disentanglement testing Sprites dataset. *https://github.com/deepmind/dsprites-dataset/*, 2017.

[38] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971.

[39] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, (115):211–252, 2015.

[40] Shengxin Qian. R-MONet: Region-Based Unsupervised Scene Decomposition and Representation via Consistency of Object Representations. *OpenReview.Net*, 2021.

[41] I. Sobel and G. Feldman. A 3x3 Isotropic Gradient Operator for Image Processing. *Pattern Classification and Scene Analysis*, pages 271–272, 1973.

[42] E. S. Spelke, K. Breinlinger, J. Macomber, and K. Jacobson. Origins of knowledge. *Psychological Review*, 99(4):605–632, 1992.

[43] K. Stelzner, R. Peharz, and K. Kersting. Faster attend-infer-repeat with tractable probabilistic models. *ICML*, 2019.

[44] S. van Steenkiste, K. Kurach, J. Schmidhuber, and S. Gelly. Investigating object compositionality in Generative Adversarial Networks. *Neural Networks*, 2020.

[45] J. von Kügelgen, I. Ustyuzhaninov, P. Gehler, M. Bethge, and B. Schölkopf. Towards causal generative scene models via competition of experts. *ICLR Workshops*, 2020.

[46] A. Voynov, S. Morozov, and A. Babenko. Object Segmentation Without Labels with Large-Scale Generative Models. *ICML*, 2021.

[47] W. Wang, Q. Lai, H. Fu, J. Shen, H. Ling, and R. Yang. Salient Object Detection in the Deep Learning Era: An In-Depth Survey. *TPAMI*, 2021.

[48] M. Wertheimer. Untersuchungen zur Lehre yon der Gestalt. *Psychologische Forschung*, (4):301–350, 1923.

[49] K. Xu, C. Li, J. Zhu, and B. Zhang. Multi-object generation with amortized structural regularization. *NeurIPS*, 2019.

[50] J. Yuan, T. Chen, B. Li, and X. Xue. Compositional Scene Representation Learning via Reconstruction: A Survey. *arXiv:2202.07135*, 2022.

[51] J. Yuan, B. Li, and X. Xue. Generative modeling of infinite occluded objects for compositional scene representation. *ICML*, 2019.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section 5
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [No] Our main contribution is the investigation and analysis on existing approaches.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See appendix
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No] On the one hand, it would be too computationally expensive. On the other hand, it has been studies extensively in the methods we are investigating.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See appendix
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]