

InstSynth: Instance-wise Prompt-guided Style Masked Conditional Data Synthesis for Scene Understanding

Thanh-Danh Nguyen^{1,2}, Bich-Nga Pham^{1,2}, Trong-Tai Dam Vu^{1,2}, Vinh-Tiep Nguyen^{†1,2},
Thanh Duc Ngo^{1,2}, and Tam V. Nguyen³

¹University of Information Technology, Ho Chi Minh City, Vietnam

²Vietnam National University, Ho Chi Minh City, Vietnam

³University of Dayton, Dayton, OH 45469, United States

{*danhnt, ngaptb, taidvt, tiepnv, thanhnd*}@uit.edu.vn, *tamnguyen@udayton.edu*, [†]*corresponding author*

Abstract—Scene understanding at the instance level is an essential task in computer vision to support modern Advanced Driver Assistance Systems. Solutions have been proposed with abundant annotated training data. However, the annotation at the instance level is high-cost due to huge manual efforts. In this work, we solve this problem by introducing InstSynth, an advanced framework leveraging instance-wise annotations as conditions to enrich the training data. Existing methods focused on semantic segmentation via using prompts to synthesize image-annotation pairs, facing an unrealistic manner. Our proposals utilize the strength of such large generative models to synthesize instance data with prompt-guided and mask-based mechanisms to boost the performance of the instance-level scene understanding models. We empirically improve the performance of the latest instance segmentation architectures of FastInst and OneFormer by 14.49% and 11.59% AP, respectively, evaluated on the Cityscapes benchmark. Accordingly, we construct an instance-level synthesized dataset, dubbed IS-Cityscapes, with over a 4× larger number of instances in comparison with the vanilla Cityscapes. Code can be found at <https://github.com/danhntd/InstSynth>.

Index Terms—Scene Understanding, Instance Segmentation, Conditional Image Generation, Prompt-based, Mask-based.

I. INTRODUCTION

In recent years, the Advanced Driver Assistance Systems (ADAS) has gradually become a vital standard system for the released automobiles in the market. ADAS is the combination of a main computing unit taking inputs from various sensors, cameras, and the vehicle driving system then outputs the necessary actions to support the driver or intervene in the driving process to prevent accidents. The computer vision system also contributes to the ADAS as a scene-understanding component. The scene understanding at the instance level is such an intense task that the models should have the ability to recognize each individual semantic instance. To perform such a task, the deep learning model should be well-trained on fine-grained annotated data of instance segmentation. However, the issue turns out when the annotation process digests a huge amount of manual effort, especially for the task needing instance-wise pixel-level annotation. This challenge occurs despite the ability to collect abundant data from dashcams, only due to the lack of detailed annotation. This phenomenon raises the requirement for proposing automatic methods that can generate data or utilize the existing annotated data to fine-tune the scene understanding models. Besides, such methods

like SoRA [1], DALL-E2 [2], Imagen [3], GLIGEN [4] or GPT series [5] are popular in the computer science community in the aspect of generating data at high quality. The question is whether we can leverage those generative models to support the paired image annotation in the instance segmentation task.

In this paper, we utilize existing annotated instance segmentation training data to create more diverse data. The existing method of Dataset Diffusion [6] solved semantic segmentation by generating image-annotations pairs based on prompt conditions. This approach faced the problem of unrealistic data due to the complete generation image. Our goal is to boost the accuracy performance of the instance-level scene understanding models with the help of synthesized instances integrated with real images. Indeed, we formulate the task as a prompt-guided mask-based conditional image generation problem to generate data for instance segmentation. Accordingly, a pair of an image and an instance-level mask in collaboration with its category-driven prompt are used to create more samples via an image generation model. The synthesized image containing the considerate instance corresponds to the masked shape of the annotation, and thus, can leverage the same mask annotation to serve the instance segmentation training pipeline. By this means, we automatically increase the diversity of the training data via the generative models. Furthermore, we reuse the limited annotation masks to save the manual annotating efforts.

Our contributions in this work are three-fold:

- Firstly, we introduce InstSynth - an instance-wise prompt-guided style masked conditional data synthesis approach for instance scene understanding. The advanced framework can make use of existing annotated data to boost the performance of the instance segmentation models.
- Secondly, we construct an instance-level synthesized dataset, dubbed IS-Cityscapes as a result of our conditional image generation method. In detail, we increase the number of instances in the training data four times more than the original [7].
- Thirdly, we empirically demonstrate the performance of our proposed method over the state-of-the-art baselines of FastInst [8] and OneFormer [9] by 14.49% and 11.59% AP, respectively, evaluated on the Cityscapes benchmark.

The rest of this paper is organized as follows. Section II

reviews related work on scene understanding, and conditional image synthesis approach. Section III presents our proposed method - InstSynth with details on each proposed component. In Section IV, we report extensive experiments and ablation studies to prove the effectiveness of our proposal. Finally, Section V concludes our work.

II. RELATED WORK

A. Urban Scene Understanding Research

The demand for applications like autonomous driving, surveillance systems, and object tracking is escalating, especially in urban scenarios. To facilitate these applications, techniques such as semantic segmentation and instance segmentation are employed, offering valuable insights into complex urban scenes.

Semantic Segmentation. Traditional methods primarily relied on CNNs, treated as a pixel classification problem [10]. Recent works [11], [12] have demonstrated the success of transformer-based methods in semantic segmentation, leveraging achievements in vision and language domains [13], [14]. While semantic segmentation effectively classifies regions into different classes, its application in urban scene tasks faces limitations. These include difficulties in accurately identifying individual objects within clusters of similar classes and the absence of clear boundaries between distinct instances in images. These challenges pose significant risks for real-world urban applications.

Instance Segmentation. Unlike semantic segmentation, instance segmentation provides more comprehensive information by classifying each individual instance at the pixel level, with clear boundaries between instances of the same and different classes. This effectively addresses urban tasks that require high-detail information. Therefore, in developing new datasets for the urban scene domain, we prioritize instance segmentation. Such methods introduced for image understanding at the instance level can be adapted to serve urban scene analysis, i.e. two-stage approach [15]–[17], and one-stage approach [8], [9], [18]–[20]. Recently, OneFormer [9] is introduced with an all-in-one manner that solves the task of panoptic, semantic, and even instance segmentation. FastInst [8] focuses on real-time applications built on top of Mask2Former [19]. Its keys include instance activation-guided queries, dual-path update strategy, and ground truth mask-guided learning. In our work, we choose FastInst [8] and OneFormer [9] as baselines to evaluate the proposals.

B. Conditional Image Synthesis Approach

Recently, numerous innovative methods have emerged as pivotal in producing high-quality and varied images followed by conditions [2], [3], [21], [22]. One such breakthrough came with the introduction of the Latent Diffusion Model [22] in the research community. The technique facilitates the input of uni- or multi-controls, such as text, masks, sketches, etc., for inpainting tasks. Therefore, there are numerous models developed based on this model including DiffInpainting [22],

TABLE I
STATISTIC ON THE NUMBER OF INSTANCES: ORIGINAL CITYSCAPES [7]
VS. OUR AUGMENTED DATA - IS-CITYSCAPES.

Inst. ID	Inst. Label	#Original Inst.	#Augmented Inst.
11	Person	17,919	69,932
12	Rider	1,781	6,950
13	Car	26,963	118,609
14	Truck	484	2,210
15	Bus	379	1,704
16	Train	168	789
17	Motorcycle	737	2,945
18	Bicycle	3,675	13,896
Total		52,106	217,035

GLIGEN [4], BlendedDiff [23], or Inst-Inpainting [24]. DiffInpainting [22] offers precise control but struggles with complex images, while GLIGEN [4] produces high-resolution outputs without considering context, sometimes leading to unrealistic results. BlendedDiff [23] optimizes Latent Diffusion Models [22] for accurate image reconstruction, particularly for local edits with thin masks, albeit with potential subtle changes. Inst-Inpainting [24] focuses on object removal tasks during training. We notice the strengths of these methods and utilize them to support our instance synthesis process.

C. Data Enhancement in Image Understanding

Traditional image data enhancement techniques rely on low-level feature augmentations, such as geometric transformations and color space transformations. Such methods may fall short in producing diverse samples that fully capture the spectrum of variations in the original dataset. This limitation can narrow the diverse range of instances in urban scenes when synthesizing new data. Recent research addresses these limitations by exploring deep learning-based augmentation techniques. These methods focus on preserving the semantic meaning of the data while increasing the image diversity through sophisticated transformations. Techniques such as data generation [22], and data embedding [25], [26] have gained popularity for enriching datasets with meaningful variations while maintaining semantic integrity. Our InstSynth framework utilizes controllable instance augmentation with diffusion-based models, leveraging the power of recent data generation methods to control the semantic factor of the generated instances.

D. Urban Scene Datasets

To evaluate the segmentation models in the urban domain, several benchmarks were proposed as standards for the community. Cityscapes [7] consists of 5,000 images of urban scenes with high-resolution pixel-level annotations of $2,048 \times 1,024$, separated into 19 semantic categories and 8 instance categories. CamVid [27] serves the semantic segmentation as a small urban dataset with around 700 images divided into 32 classes. Apart from the aforementioned ones, Vistas Mapillary [28] is a huge dataset introduced with up to 25K images at high resolution and is annotated into 66 categories. Notably, the task of instance segmentation faces a limited number of supporting datasets. To demonstrate our proposed

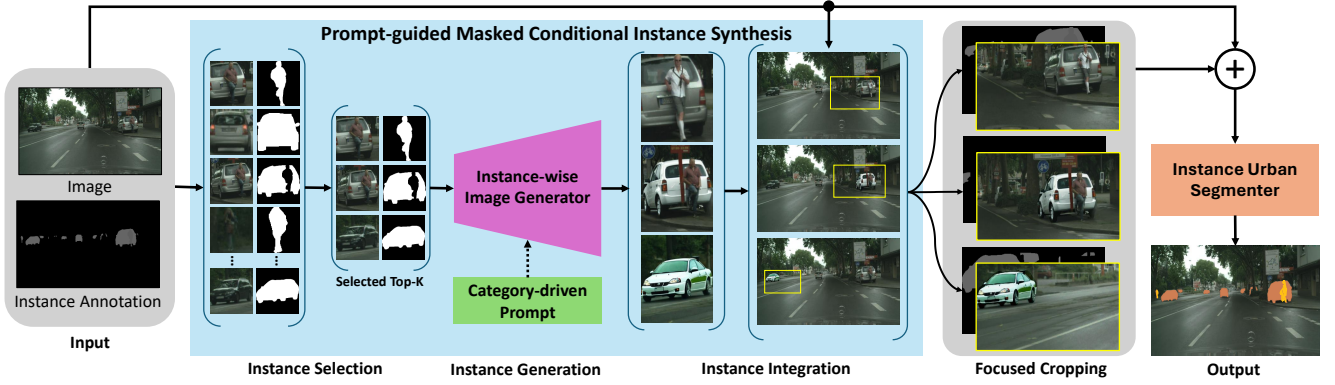


Fig. 1. Overview of our InstSynth framework. The pipeline allows a pair of image-annotation to be augmented into various variations with category-driven text prompts in terms of boosting the data diversity to serve instance urban scene understanding.

idea on instance-wise data generation, we consider Cityscapes [7] as a potential benchmark for this work due to the various types of provided annotations.

III. PROPOSED METHOD

A. Overview our InstSynth framework

Our proposed InstSynth framework, as illustrated in Fig. 1, is recognized with the instance-wise image synthesis process to serve the scene understanding task. We design a data synthesis mechanism to augment the input urban scene at the instance level with mask conditions and prompt guidance. This stage enhances the diversity of any specific instances to strengthen the generalization of the segmentation model. Accordingly, an image-instance annotation pair is taken into the process of data synthesis along with the support of a category-driven prompt. The results of this process are different variants of augmented instances of the single input image. Altogether, the original and its augmented versions are combined to train the instance segmentation model. We discuss the details of the framework in the following sections.

B. Prompt-guided Masked Conditional Instance Synthesis

This section outlines our innovative method for synthesizing urban images from the Cityscape dataset [7], contributing to the generation of realistic urban scenes while adhering strictly to dataset regulations and reliability standards.

1) **Phase 1 - Instance Preparation:** In the initial phase of our synthesis method, the first step is instance selection and cropping from mask annotations in the Cityscapes training set. This foundation step involves the identification of prominent instances I within the dataset images, specifically within the instance mask annotation, denoted as M . Employing an algorithmic strategy, as described in Eqn. 1, we discern the top K instances, where K is empirically predetermined (e.g., $K = 3$ in our implementation), based on their size quantified by the pixel count within each mask. Subsequently, these selected instances undergo a cropping process while concurrently eliminating any subsidiary instances, thereby ensuring that the generating process focuses solely on the selected instances. These instances are cropped with a fixed ratio of 1 : 1. Simultaneously, the corresponding natural images, denoted as N , undergo identical cropping actions. These cropped natural

images and masks serve as essential inputs for the subsequent image synthesizing process, aimed at seamlessly filling the masked regions within the instance images.

$$I_N(K) = \max_{v \in V} \sum_{i=1}^h \sum_{j=1}^w \delta(M[i, j], v), \quad (1)$$

where:

- h and w are the height and width of mask M
- K is the number of selected instances
- $M[i, j]$ is the pixel value at position $[i, j]$ in M
- v is a component of the unique values list V of M
- $\delta(x, y)$ is the *Kronecker* delta function, defined as:

$$\delta(x, y) = \begin{cases} 1, & \text{if } x = y \\ 0, & \text{otherwise} \end{cases}$$

2) **Phase 2 - Instance Generation:** In this stage, the instance images derived from Phase 1 serve as the primary input. The overarching objective of this phase is to effectively complete the images by inferring and generating instance content to fill the masked regions within the original images. We aim to generate new instances from existing ones to enrich our data while ensuring that the existing masks remain useful. Utilizing state-of-the-art image-generation models of GLIGEN [4], DiffInpainting [22], and BlendedDiff [23] with Stable Diffusion XL (SDXL) [29], we create new instances from instance images paired with corresponding masks and pre-defined category-driven prompts.

More specifically, we define the category-driven prompt-guided styling factors using two structures: “A photo of a [color] [instance category]” for transports (e.g., “A photo of a red car”), and “A photo of [a man/a woman]” for people [6]. Despite its simplicity, this approach proves invaluable in diversifying our generated instances, effectively advancing our objectives. In our initial experiments with those models, we observed promising results where generated instances nearly substituted the input instances in their proper positions while maintaining a realistic appearance and adhering to the prompt. Fig. 2 illustrate exemplary results of our methods. GLIGEN [4] demonstrates remarkable performance in terms of visually coherent and semantically meaningful contents by accurately filling in regions identified by masks with almost exact instances and subsequently generating high-resolution outputs

TABLE II
COMPARISON ON FASTINST BASELINES [8] ON CITYSCAPES
VALIDATION SET [7].

Method	Backbone	Generation Base	AP	AP50
Mask2Former† [19]	R50-FPN-D3†	-	31.40	55.90
FastInst [8]	R50-FPN-D3†	-	35.50	59.00
	R50-FPN-D3*	-	24.93	45.69
	R50-FPN-D3**	-	27.65	49.21
InstSynth (Ours)	FastInst- R50-FPN-D3**	GLIGEN [4]	34.88	59.20
		DiffInpainting [22]	36.44	62.06
		BlendedDiff [23]	36.52	62.21

† denotes the published results of [8]

* denotes our reproduced results of FastInst w/o CLIP

** denotes our reproduced results of FastInst w/o CLIP, and w/ customized image sizes

The first, second, and third best results are marked in red, blue, and green, respectively.

while DiffInpainting [22] could preserve the nearly original resolution. Thanks to the blending mechanism of BlendedDiff [23], it effectively fills in regions with the instances generated by SDXL. Since all selected models produced satisfactory results, we opted to utilize them under three different versions to serve the instance segmentation models.

3) Phase 3 - Instance Integration and Focused Cropping:

The final phase of our synthesis pipeline encompasses the integration of inpainted images back into the original natural images to create a new set of images with augmented instances, denoted as N , guided by the corresponding masks M derived in Phase 1. Additionally, to generate images that emphasize the instances generated in Phase 2, we design a focused cropping algorithm. This involves cropping the images with a fixed 2 : 1 aspect ratio, aligned with the original image ratio of the Cityscapes dataset [7] (i.e. 2048×1024 pixels), thereby preserving the proportions of the instances in the images. The cropping ratio is meticulously determined using a normal distribution, facilitating the judicious allocation of space to accentuate the instances within the cropped images. The instances may appear randomly in the images, thus ensuring the diversity of the dataset. The formula for selecting a random ratio from the normal distribution to calculate the area of the new image region, denoted as S_N , is described in Eqn. 2. From this calculation, the height h_{target} and width w_{target} dimensions of the new image with the fixed 2 : 1 ratio are derived, and the new images are cropped based on the computed coordinates $\text{Coord}(S_N)$ (shown in Eqn. 3). The outcome of this process is the generation of sets of images, each focusing on a selected instance, while maintaining the dataset regulations and standards.

$$S_N = \lambda S_{\text{ins}}, \lambda \sim \mathcal{N}(\mu, \sigma) \quad (2)$$

$$\text{Coord}(S_N) = \begin{cases} x_{\text{target}} \in [\max(0, x_{\text{ins_min}} - (w_{\text{target}} - w_{\text{ins}})), x_{\text{ins_min}}] \\ y_{\text{target}} \in [\max(0, y_{\text{ins_min}} - (h_{\text{target}} - h_{\text{ins}})), y_{\text{ins_min}}] \\ h_{\text{target}} = w_{\text{target}}/2 = S_N/4 \end{cases} \quad (3)$$

where:

- λ follows a normal distribution \mathcal{N} with $\mu = 3$, $\sigma = 1$
- S_{ins} is the area of the instance region

By completing this step, we contribute an augmented dataset version with synthesized images at the instance level, named after IS-Cityscapes. Tab. I provides a statistical report to compare the number of images among versions. Our proposed methods successfully create a four times larger number of instances per category compared to the vanilla dataset.

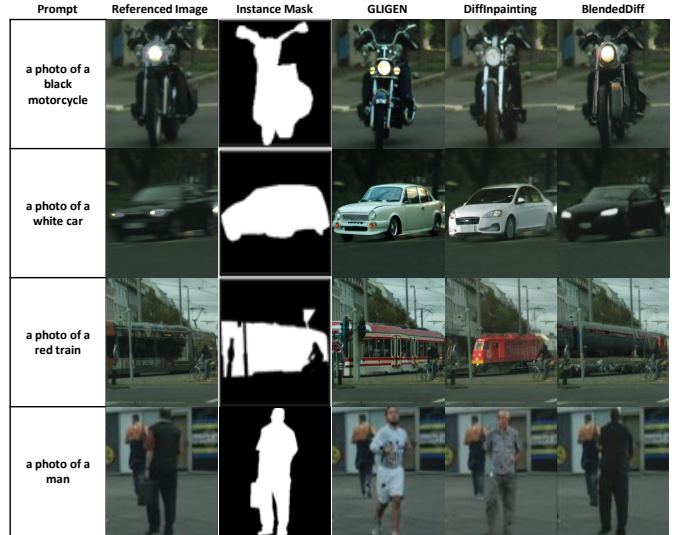


Fig. 2. Exemplary instance image generation from different models of GLIGEN [4], DiffInpainting [22], and BlendedDiff [23]. We present the referenced images and their corresponding prompts to synthesize instances.



Fig. 3. Visualization results on Cityscapes val-set [7] with our FastInst R50-FPN-D3 [8]. The confidence threshold is 0.8. Best viewed with zoomed-in.

C. Instance-wise Urban Segmenter

In our InstSynth, we employ FastInst [8] and OneFormer [9] as baselines to perform instance-wise urban scene understanding tasks. FastInst [8] prioritize the real-time manner with a light-weight pixel-decoder component. On the contrary, OneFormer [9] is an all-in-one model with the task-conditioned joint training strategy that can train on ground truths of a single domain (i.e. semantic, instance, or panoptic segmentation) within a single multi-task training process. We establish these two models as examples to demonstrate the effectiveness of our proposed framework. To this end, the instance urban segmenter digests annotated data from a real image and its augmented versions to perform model training.

IV. EXPERIMENTS

A. Experimental Configurations

Dataset. Our experiments leverage the Cityscapes dataset [7] with its instance-level annotation as the main training data. We perform the evaluation process on the public validation set of this benchmark which includes 500 images distributed into all the instance classes.

Experimental Settings. In terms of instance segmentation, we experimented on the two baselines of FastInst [8] and OneFormer [9]. Adapting to the limitations of our hardware computing units, we reduce the original configurations to a smaller crop size of 360×720 instead of $512 \times 1,024$ (FastInst [8]) or 512×512 (OneFormer [9]). Notably, our models are

TABLE III
STATE-OF-THE-ART COMPARISON ON ONEFORMER [9] WITH ADAPTIVE CROP-SIZE EVALUATED ON CITYSCAPES VALIDATION SET [7].

Method	Backbone	Version	Crop size	PQ \uparrow	IoU \uparrow	AP \uparrow	AP50 \uparrow
CMT-DeepLab \ddagger [30]	MaX-S \ddagger [30]	-	1025 \times 2049	64.60	81.40	-	-
Axial-DeepLab-L \ddagger [31]	Axial ResNet-L \ddagger [31]	-	1025 \times 2049	63.90	81.00	35.80	-
Axial-DeepLab-XL \ddagger [31]	Axial ResNet-XL \ddagger [31]	-	1025 \times 2049	64.40	80.60	36.70	-
Panoptic-DeepLab \ddagger [32]	SWideRNet \ddagger [33]	-	1025 \times 2049	66.40	82.20	40.10	-
OneFormer [9]	Mapillary-ConvNext-L Swin-L	Original	360 \times 720	48.84	72.58	21.75	40.94
			360 \times 720	51.52	74.53	25.68	45.90
InstSynth* (Ours)	Mapillary-ConvNext-L Swin-L	GLIGEN [4]	360 \times 720	62.90	80.55	38.46	64.73
			360 \times 720	60.33	79.18	35.67	61.09
	Mapillary-ConvNext-L Swin-L	DiffInpainting [22]	360 \times 720	62.90	80.96	38.66	64.69
			360 \times 720	60.13	77.88	35.40	60.50
	Mapillary-ConvNext-L Swin-L	BlendedDiff [23]	360 \times 720	63.33	80.88	38.93	64.91
			360 \times 720	60.47	79.10	35.75	61.01

* denotes our methods based on OneFormer instance segmentation architecture

All of our reproduced results of OneFormer are w/o CLIP, and w/ smaller crop size

The first, second, and third best results are marked in red, blue, and green, respectively.

trained without a CLIP-based backbone, which is the main factor raising the training memory. Our framework is built on top of Detectron2 [34] and the configurations of other models originated from their publications. In detail, we adopted four GeForce RTX 2080Ti GPUs and trained with the AdamW optimizer. Our training process occurred with 90K iterations with a batch size of 4, and the base learning rate of $1e^{-4}$. Besides, we follow the published settings of [4], [22], [23] correspondingly during the generation processes.

Evaluation metrics. To report our instance segmentation results, we use average precision (AP). In detail, we report AP and AP@50. Please reach this site <https://cocodataset.org/#detection-eval> for details on the evaluation metrics. In the case of OneFormer [9], we also report Panoptic Quality (PQ) and Intersection-over-Union (IoU) to evaluate panoptic and semantic segmentation. To measure the quality of the generated images, we choose the four common metrics of CLIPScore [35], FID [36], SSIM [37], and PSNR [38]. Namely, PSNR evaluates the fidelity of generated images compared to the original images; SSIM assesses structural information and perceived similarity between the original and generated images; CLIP-Score evaluates how well-generated images align with the original images, particularly regarding object labels; and FID quantifies the realism and diversity of images.

B. State-of-the-art Image Understanding Comparison

We reported the established experiments on our InstSynth framework in Tab. II and Tab. III. We also rely on the published work of [19], [30]–[32] to compare our results on adaptive configurations. For a fair comparison, we noticed the same backbone architectures, i.e. R50-FPN-D3 for FastInst [8], Mapillary-ConvNext-L and Swin-L for OneFormer [9]. To this end, we improve a large margin compared to the chosen baselines. In detail, our FastInst-based instance segmenter achieves 34.88%, 36.44%, and 36.52%AP via GLIGEN [4], DiffInpainting [22], and BlendedDiff [23], respectively. These APs improve over 7.23%, 8.79%, and 8.87% compared to our FastInst reproduced results without CLIP, and with customized training image sizes. To this end, we observe the effectiveness

TABLE IV
QUANTITATIVE COMPARISON OF IMAGE GENERATOR INCLUDING GLIGEN [4], BLENDEDIFF [23], AND DIFFINPAINTING [22]

Method	CLIPScore \uparrow	FID \downarrow	SSIM \uparrow	PSNR \uparrow
GLIGEN [4]	0.79	40.65	0.67	14.39
DiffInpainting [22]	0.81	31.03	0.72	15.95
BlendedDiff [23]	0.87	16.28	0.90	25.23

The best results are marked in bold.

of BlendedDiff [23] in generating conditional instance-level data. Regarding the OneFormer-based model, we increase a large margin of AP compared among the baselines, i.e. from 21.75% to approx. 38.93% on ConvNext-L and from 25.68% to approx. 35.75% on Swin-L backbones. To adapt the hardware availability, we reduced the crop size of the input training image down to 360 \times 720, which affected our panoptic and semantic segmentation performance measured on PQ and IoU. However, as focusing on the instance segmentation task, we achieve comparable AP compared to other methods. Meanwhile, we demonstrate our proposals work well when compared with our baselines. Exemplary visualization is provided in Fig. 3. We provide further direct investigation on the impact of instance synthesis results in the next section.

C. Instance-wise Synthesis Ablation Evaluation

Besides the indirect evaluation based on the segmentation methods, we report Tab. IV with the quantitative comparison on common metrics [35]–[38] to analyze the impact of image synthesizing methods. In our case, BlendedDiff [23] yields the highest performance over the four mentioned metrics. The results can be explained by two main reasons. Firstly, BlendedDiff [23] minimally modifies the original images as to the blending process. Secondly, prevailing evaluation metrics within this domain predominantly qualify generated images with their corresponding originals. While DiffInpainting [22] and GLIGEN [4] demonstrate noteworthy outcomes, their effectiveness is compromised due to their neglect of the contextual and resolution aspects surrounding the masked regions.

Discussion. In Fig. 4, we present several failure cases in our instance image generation, where the instances are intense, i.e. instances with small details, similar texture with background,









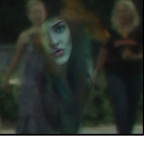
Prompt	Mask	Referenced Image	Synthesized Image
a photo of a black bicycle			
a photo of a black train			
a photo of a woman			

Fig. 4. Failure cases when generating intense instances, i.e. instances with small or similar details, strange masked shapes, or high density.

strange masked shape, or high density. In those cases, we observe the limitations of our conditional image generator when the referenced masks are too complicated to guide the model, or the category-driven prompts are too simple to depict the instance image. Potential solutions can be proposed based on these two aforementioned points.

V. CONCLUSION

In this paper, we propose InstSynth - an instance-wise prompt-guided style masked conditional data synthesis approach for instance-wise scene understanding. The framework utilizes existing annotated data to boost the performance of the instance segmentation models. We address the annotation data limitation at instance-wise pixel-level details of the instance segmentation task. Furthermore, we contribute IS-Cityscapes which is a synthesized urban instance segmentation dataset. Via our instance generation approach, we increase four times the number of instances in the original dataset, resulting in over 200K urban instances serving the training process. Finally, we conduct extensive experiments and ablation investigations to prove the effectiveness of our methods over the latest architectures. In the future, we plan to improve the ability of our instance generation method to deal with various diversity to solve real-world intense situations while driving.

VI. ACKNOWLEDGEMENT

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number DS2024-26-06

REFERENCES

- [1] Y. Liu, K. Zhang, Y. Li, Z. Yan, C. Gao, R. Chen, Z. Yuan, Y. Huang, H. Sun *et al.*, “Sora: A review on background, technology, limitations, and opportunities of large vision models,” *arXiv preprint*, 2024.
- [2] A. Ramesh, P. Dhariwal, A. Nichol *et al.*, “Hierarchical text-conditional image generation with clip latents,” *arXiv preprint*, 2024.
- [3] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, *et al.*, “Photorealistic text-to-image diffusion models with deep language understanding,” *NeurIPS*, vol. 35, pp. 36479–36494, 2022.
- [4] Y. Li, H. Liu, Q. Wu, F. Mu, J. Yang, J. Gao, C. Li, and Y. J. Lee, “Gligen: Open-set grounded text-to-image generation,” in *CVPR*, 2023.
- [5] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altschmidt, S. Altman, S. Anadkat *et al.*, “Gpt-4 technical report,” *arXiv preprint*, 2023.
- [6] Q. Nguyen, T. Vu, A. Tran, and K. Nguyen, “Dataset diffusion: Diffusion-based synthetic data generation for pixel-level semantic segmentation,” *NeurIPS*, vol. 36, 2023.
- [7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson *et al.*, “The cityscapes dataset for semantic urban scene understanding,” in *CVPR*, 2016, pp. 3213–3223.
- [8] J. He, P. Li, Y. Geng, and X. Xie, “Fastinst: A simple query-based model for real-time instance segmentation,” in *CVPR*, 2023, pp. 23663–23672.
- [9] J. Jain, J. Li, M. T. Chiu, A. Hassani, N. Orlov, and H. Shi, “Oneformer: One transformer to rule universal image segmentation,” in *CVPR*, 2023.
- [10] B. Cheng, L.-C. Chen, Y. Wei, Y. Zhu, Z. Huang, J. Xiong, T. Huang, W.-M. Hwu, and H. Shi, “Spgnet: Semantic prediction guidance for scene parsing,” in *CVPR*, 2019.
- [11] R. Strudel, R. Garcia, I. Laptev, and C. Schmid, “Segmenter: Transformer for semantic segmentation,” in *ICCV*, 2021.
- [12] E. Xie, W. Wang, Z. Yu, A. Anandkumar, J. M. Alvarez, and P. Luo, “Segformer: Simple and efficient design for semantic segmentation with transformers,” in *NeurIPS*, 2021.
- [13] N. Carion, F. Massa, G. Synnaeve, N. Usunier *et al.*, “End-to-end object detection with transformers,” in *ECCV*, 2020.
- [14] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” *NeurIPS*, 2017.
- [15] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in *ICCV*, 2017, pp. 2980–2988.
- [16] Z. Cai and N. Vasconcelos, “Cascade r-cnn: Delving into high quality object detection,” in *CVPR*, 2018.
- [17] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, “Path aggregation network for instance segmentation,” in *CVPR*, 2018.
- [18] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, “Yolact: Real-time instance segmentation,” in *ICCV*, 2019.
- [19] B. Cheng, I. Misra, A. G. Schwing, A. Kirillov, and R. Girshick, “Masked-attention mask transformer for universal image segmentation,” in *CVPR*, 2022, pp. 1290–1299.
- [20] T.-D. Nguyen, D.-T. Luu, V.-T. Nguyen, and T. D. Ngo, “Ce-ost: Contour emphasis for one-stage transformer-based camouflage instance segmentation,” in *MAPR*, 2023, pp. 1–6.
- [21] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss *et al.*, “Zero-shot text-to-image generation,” in *ICML*. Pmlr, 2021, pp. 8821–8831.
- [22] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, “High-resolution image synthesis with latent diffusion models,” in *CVPR*, 2022.
- [23] O. Avrahami, D. Lischinski, and O. Fried, “Blended diffusion for text-driven editing of natural images,” in *CVPR*, 2022.
- [24] A. B. Yildirim, V. Baday, E. Erdem, A. Erdem, and A. Dundar, “Inst-inpaint: Instructing to remove objects with diffusion models,” 2023.
- [25] C.-W. Kuo, C.-Y. Ma, J.-B. Huang, and Z. Kira, “Featmatch: Feature-based augmentation for semi-supervised learning,” in *ECCV*, 2020.
- [26] R. Volpi, P. Morerio, S. Savarese, and V. Murino, “Adversarial feature augmentation for unsupervised domain adaptation,” in *CVPR*, 2018.
- [27] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” in *Pattern Recognition Letters*, vol. 30, no. 2. Elsevier, 2009, pp. 88–97.
- [28] G. Neuhold, T. Ollmann *et al.*, “The mapillary vistas dataset for semantic understanding of street scenes,” in *ICCV*, 2017.
- [29] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, “Sdxl: Improving latent diffusion models for high-resolution image synthesis,” *arXiv preprint*, 2023.
- [30] Q. Yu, H. Wang, D. Kim, S. Qiao *et al.*, “Cmt-deeplab: Clustering mask transformers for panoptic segmentation,” in *CVPR*, 2022.
- [31] H. Wang, Y. Zhu, B. Green *et al.*, “Axial-deeplab: Stand-alone axial-attention for panoptic segmentation,” in *ECCV*, 2020.
- [32] B. Cheng, M. D. Collins *et al.*, “Panoptic-deeplab: A simple, strong, and fast baseline for bottom-up panoptic segmentation,” in *CVPR*, 2020.
- [33] L.-C. Chen, H. Wang, and S. Qiao, “Scaling wide residual networks for panoptic segmentation,” *arXiv preprint*, 2020.
- [34] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, “Detectron2,” <https://github.com/facebookresearch/detectron2>, 2019.
- [35] J. Hessel, A. Holtzman, M. Forbes *et al.*, “Clipscore: A reference-free evaluation metric for image captioning,” in *EMNLP*, 2021.
- [36] M. Heusel, H. Ramsauer *et al.*, “Gans trained by a two time-scale update rule converge to a local nash equilibrium,” *NeurIPS*, 2017.
- [37] Z. Wang, A. C. Bovik, H. R. Sheikh *et al.*, “Image quality assessment: From error visibility to structural similarity,” *IEEE TIP*.
- [38] A. Hore and D. Ziou, “Image quality metrics: Psnr vs. ssim,” in *ICPR*. IEEE, 2010, pp. 2366–2369.