S^5: Sketch-to-image Synthesis via Scene and Size Sensing

Samah S. Baraheem
Department of Computer Science
Umm Al-Qura University
Makkah, Saudi Arabia
ssbaraheem@uqu.edu.sa

Tam V. Nguyen
Department of Computer Science
University of Dayton
Dayton, USA
tnguyen1@udayton.edu

***Abstract*— Sketch-to-image synthesis method transforms a simple abstract black-and-white sketch into an image. Most sketch-to-image synthesis methods generate an image in an end-to-end manner, leading to generate a non-satisfactory result. The reason is that, in end-to-end models, the models generate images directly from the input sketches. Thus, with very abstract and complicated sketches, the models might struggle in generating naturalistic images due to the simultaneous focus on both factors: overall shape and fine-grained details. In this paper, we propose to divide the problem into subproblems. To this end, an intermediate output, which is a semantic mask map, is first generated from the input sketch via an instance and semantic segmentation. In the instance segmentation stage, the objects' sizes might be modified depending on the surrounding environment and their respective size prior to reflect reality and produce more realistic images. In the semantic seg-mentation stage, a background segmentation is first constructed based on the context of the detected objects. Various natural scenes are implemented for both indoor and outdoor scenes. Following this, a foreground segmentation process is commenced, where each detected object is semantically added into the constructed segmented background. Then, in the next stage, an image-to-image translation model is leveraged to convert the semantic mask map into a colored image. Finally, a post-processing stage is incorporated to further enhance the image result. Extensive experiments demonstrate the superiority of our proposed method over state-of-the-art methods.**

Keywords*— *Sketch-to-Image, Image Generation, Image Synthesis, Scene and Size Sensing.

I. INTRODUCTION

"Pictures speak louder than words." - Anonymous

Image can be seen everywhere since it conveys a story, a fact, or an imagination without any words. A single image can convey multiple meanings based on the viewers' point of view, while usually a single sentence can only maintain one meaning. In addition, images can substitute the sentences because the human brain can extract the knowledge from images faster than words. Furthermore, images have a long-lasting impression in individuals' memory. However, creating an image from scratch is not only time-consuming, but also it requires skills. Moreover, it is a painful and a tedious task. In order to generate an image in less time and without any artistic skills, sketch-to-image synthesis can be adopted. The reason is that hand sketches are much easier to produce, where only the key structural information is contained. Moreover, it can be drawn without skills and in less time. On the other hand, images contain not only the structural information and objects' boundaries but also contain other important features, including color, saturation, luminance, brightness, texture, and shadow, just to name a few. Thus, it consumes a long time and requires skills. Hence, much research has been conducted in sketch-based image generation field, where only a black and white rough sketch with key structural information is required. Then, the input sketch is automatically mapped without human intervention into the corresponding image. Therefore, anyone can create an image even without artistic skills and in no time. Different techniques have been adopted to create images from input sketches.

One research direction is sketch-based image retrieval (SBIR) systems [1]. Nonetheless, several issues might result from SBIR systems. First, fine-grained images might not be retrieved by the system due to the manual feature extraction process. In addition, SBIR might not work with poorly sketched objects since the systems may not be able to appropriately recognize the objects in the input sketch to retrieve the most similar image. Moreover, the system might not generate images that are sufficiently comparable to the input sketches, particularly in terms of orientation, perspective, or occlusion features.

To address the aforementioned problems in SBIR method, researchers have leveraged deep convolutional neural networks (CNNs) in sketch-based image synthesis task [2]. Because CNNs automatically learn the features rather than manually extracting them, CNNs-based image synthesis methods produce better images than SBIR method for the sketch-to-image problem. However, unnaturalistic image might be generated when the input sketch has several drawn objects. Image generation has become a hot topic, and much research has been conducted, especially after proposing the Generative Adversarial Network (GAN). Incorporating GAN in sketch-based image generation tasks [3-4] improves the generated results over time. However, GAN-based image synthesis

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2024.3375610

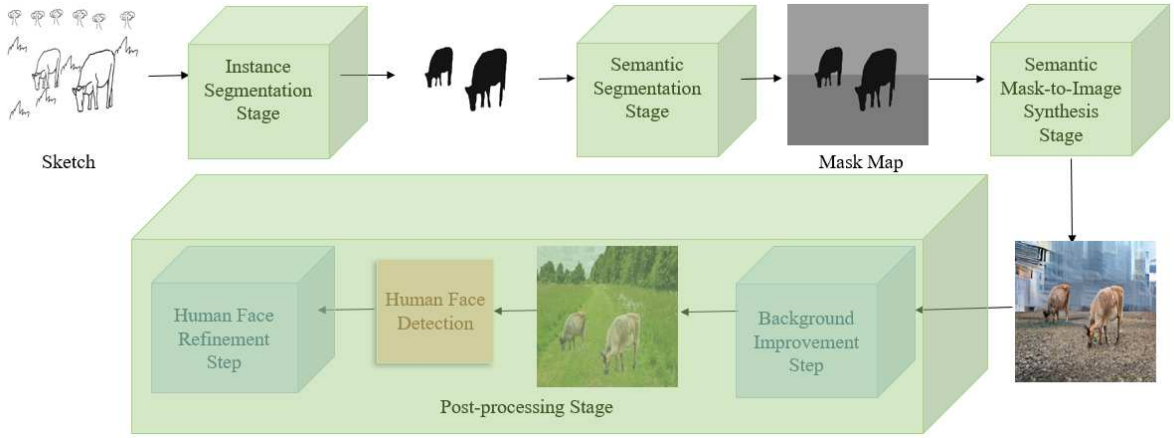


Fig. 1. The overview of our proposed sketch to image synthesis S^5 method. Our framework consists of four main stages (Instance Segmentation Stage, Semantic Segmentation Stage, Semantic Mask-to-Image Synthesis Stage, and Post-processing Stage). The fourth and last stage is composed of two steps, which are Background Improvement Step and Human Face Refinement Step to further enhance the generated images.

methods are still struggling to generate photo-realistic images from complex scenes with multiple objects.

Another technique used in image generation problem is diffusion models. Incorporating diffusion models in sketch-to-image synthesis task [5] may help with complex sketches but suffer from the abstraction and simplicity of drawn sketches. This might lead to generate unsatisfactory images.

Previous state-of-the-art sketch-to-image synthesis methods have shown great success in generating images; however, the results are still unrealistic, especially with complicated sketches. Indeed, the generated images of complex scenes with multiple objects are still a challenging problem, and the performance of the current methods is unsatisfactory. This might be because the generation process occurs in one shot, where the sketches are directly mapped into images, leading to generating unrealistic images from complex sketches.

To this end, the proposed S^5, short form of Sketch-to-image Synthesis via Scene and Size Sensing, method attempts to not only generate realistic images from complicated sketches but also reflect the reality of the objects' sizes in different environments by decomposing the problem into subproblems. It first generates intermediate outputs, namely, mask maps from the input sketches through an instance segmentation and semantic segmentation. The intermediate output maintains the boundaries, shapes, layouts, and overall structures. Then, the mask maps are mapped into colored images through image-to-image translation models, where textures, colors, shadows, among other features are preserved. To reflect reality with regard to the objects' sizes and generate more realistic images, we propose that the objects' sizes are modified based on the surrounding environment and masks' prior size. This leads to generating more photo-realistic images compared to prior models. Our contribution is as follows.

- First, using four different techniques, *i.e.*, dodging and burning [6], Holistically-nested edge detection [7], Canny edge detector and Sobel operators, we generate a sketch-like image dataset depending on MS-COCO.

- Next, leveraging our created dataset of sketch-like images, we fine-tune an instance segmentation model [8]. To reflect reality in terms of the objects' size compared to the scene, the objects' sizes might be modified based on the surrounding environment (indoor vs. outdoor). Thus, our framework first determines the background depending on the context of the existing objects, and then objects' sizes might be changed based upon a computed factor.

- Furthermore, the semantic mask segmentation works in two levels which are background segmentation and foreground segmentation.

- To further improve the synthetic images, two post-processing steps are included: background improvement step and face refinement step. In the background improvement step, 10k scene images divided into 365 different scene classes are collected. Following this and depending on the classified scene, a scene image is chosen by our method, and then the generated foregrounds are blended into the selected scene image in a specific pre-defined location so that objects are in a proper location to maintain realism. In the face refinement step, human faces are first extracted, and then, reconstructed face images are obtained via an autoencoder model, followed by aligning the reconstructed faces into the respective synthetic image.

- Finally, a dataset for evaluation purpose is compiled. This dataset is composed of 378 distinct sketch styles.

II. Related Work

A. Semantic Segmentation

Semantic segmentation is the task that identifies not only the semantic labels for each object in the image but also defines the precise regions and where each object starts and ends. It works by assigning a label for every pixel in the image. Hence, it is useful for many applications that require accurate image maps, such as autonomous driving, crowd counting, image-to-image translation, satellite imagery, medical imaging, and robotic vision. Many studies have been accomplished in this field leveraging different techniques [9-12].

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2024.3375610

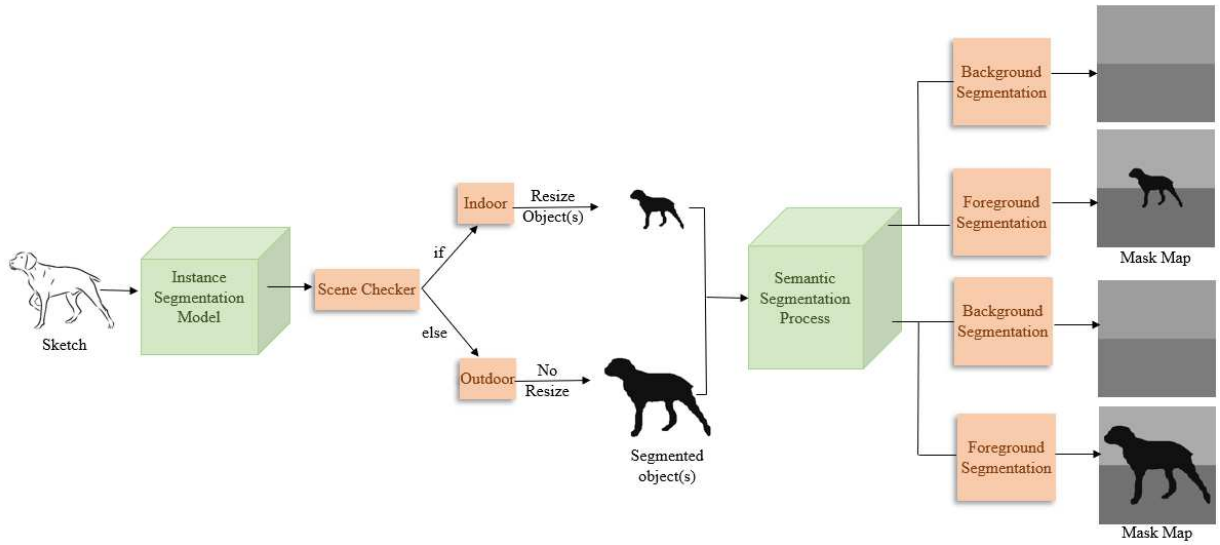


Fig. 2. The flowchart of the first two stages, namely, instance segmentation and semantic segmentation, after training the instance segmentation model on the four different edge map datasets.

B. Sketch-to-image Synthesis Methods

- **Sketch-based Image Retrieval (SBIR)**

One of first approaches is sketch-based image retrieval (SBIR) system [1]. In SBIR, a database or a search engine is leveraged to query a simple rough sketch and retrieve the most similar colored image to the corresponding input sketch. The similarity criteria between the input sketch and the corresponding colored image are determined by the descriptor. However, many problems might appear. One major problem is that it might be hard to find the appropriate matched image to the corresponding sketch. Furthermore, many problems might occur regarding the retrieved images' objects. It might retrieve an improper object orientation or improper object occlusion. Another challenge is the inability to retrieve fine-grained images due to the manual features extraction process. Additionally, with badly drawn sketches, SBIR might not work well in producing a proper respective image.

- **Sketch-to-Image via Convolutional Neural Networks (CNNs)**

Due to the limitations in SBIR and after developing CNNs, researchers have shifted their direction and used deep CNNs to map an input sketch to a colored image [2]. Unlike SBIR, where it lacks the fine-grained retrieval because the features are extracted manually via the descriptor, CNN is able to maintain the fine-grained details because it learns the feature and extracts them automatically. While the results of leveraging CNNs in sketch-based image generation are much better than the results obtained through SBIR, generated images lack high level of realism, especially with multiple objects in the input sketch.

- **Sketch-to-Image via Generative Adversarial Network (GAN)**

With the advancement in machine learning and deep learning, different models have been proposed. Recently, a popular and commonly used model has been developed, named GAN. GAN has proven its capability in image generation tasks; thus, researchers have moved to GAN to synthesize a colored image from the corresponding input sketch [3-4]. Incorporating GAN can generate realistic images from sketches. However, many challenges emerge from leveraging GAN. It may require a large dataset of sketch-photo pairs to train the model. Another limitation presented in [3] is that each class should be trained separately. Thus, it is not only time-consuming and high computation, but also less efficient. Another shortcoming is the inability to generate images from complicated sketches with fine-grained details. Furthermore, synthesized images may not be visually realistic to resemble real-world images, especially with complex sketches. Hence, to date, GAN-based sketch-to-image synthesis models produce better results than previous approaches. However, with complex sketches that consist of multiple objects, the results are still unsatisfactory.

- **Sketch-to-Image via Diffusion Model (DM)**

Many studies have leveraged diffusion models in the image synthesis field. PITI [5] uses a pretrained model that is capable of capturing the entire distribution of the natural image. This framework works well for mask-to-image and geometry-to-image translation, but it may fail with the sketch-to-image generation, especially with different sketch styles. The reason is that the model is trained on sketches extracted only via HED [7].

III. Proposed Method

Dataset: Since there is no available sketch dataset with complex scenes and multiple objects in one scene, we aim to create our own sketch dataset based on MS-COCO dataset which contains over 118k and 5k images for training and validation, respectively. Four different methods are leveraged to convert images into sketch-like images. Specifically, dodging and burning [6], Holistically-nested edge detection [7], Canny edge detector and Sobel operators are used. The reason for incorporating different types of edge maps is to improve model detection and segmentation since people tend to sketch

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2024.3375610

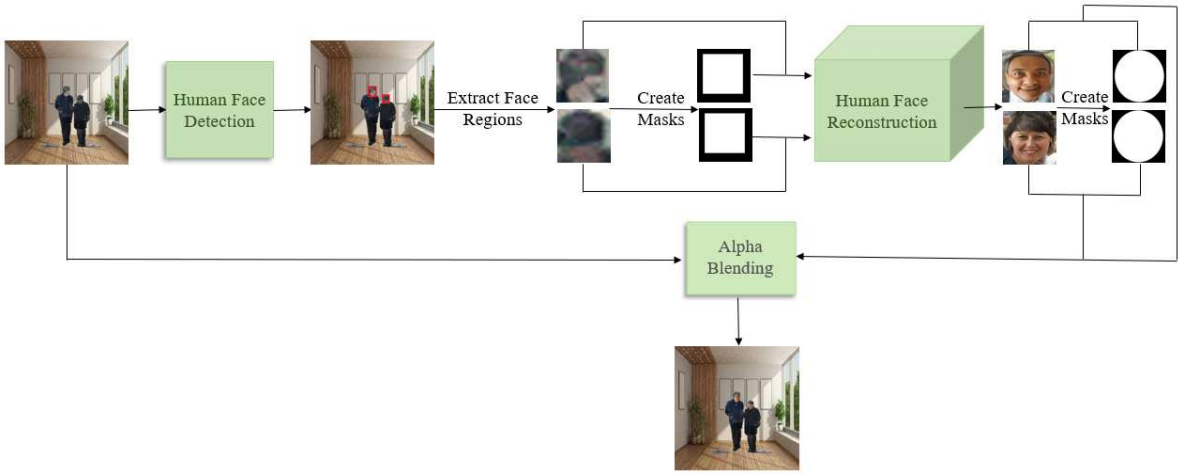


Fig. 3. An illustration of the human face refinement step. In this step, a human face detector is first used to detect and locate the human faces, followed by extracting the face regions and creating a square mask for each extracted face. Then, a new face is generated for each extracted face through an autoencoder network, namely, ICT image completion model [14]. To align the reconstructed faces, another circle mask is created for each reconstructed face, and an alpha blending technique is leveraged to align the refined face in the exact location of the generated image produced by the previous step: Background Improvement Step.

differently. Hence, having such a dataset helps in image generation process even with various sketch styles. In total, our dataset consists of over 472k and 20k different edge maps and sketches for training and validation, respectively.

Methodology: The framework is comprised of four main stages. Figure 1 illustrates the overall view of our proposed framework. Details are described as follows.

A. Instance Segmentation Stage

The instance segmentation stage starts by re-training an instance segmentation model. This model usually takes a colored image as input and outputs only the segmented objects as masks. DetectoRS [8] is leveraged as an instance segmentation model and fine-tuned for 30 epochs on our newly created dataset of sketch-like images. This stage produces a mask for each detected and segmented object in the input sketch. Meanwhile, to reflect reality regarding objects' size and maintain realistic results, objects' sizes might be modified based on the surrounding environment (indoor vs. outdoor). In our proposed method, we modify the object size if the surrounding environment is determined to be an indoor scene so that the object size is consistent with the scene. To this end, the environment/background scene is first determined based on the detected objects. To define the scene based on the existing objects' context, a simple yet effective algorithm is leveraged. Specifically, based on the prior and existing knowledge of the location of each object in real life, each object is categorized in one or more scenes. Then, for each recognized and segmented object in the input sketch, the corresponding scene(s) is increased by one. At the end of this simple yet effective algorithm, the final scene is the scene with the highest value. In total, 26 environment/background scenes are identified which are categorized into indoor and outdoor. For the indoor scenes, eight scenes are specified, which are living room, dining room, office room, child room, bedroom, kitchen, bathroom, and bookstore. In the meantime, three outdoor scene subcategories are defined for the outdoor scenes, namely, natural, transportation, and sport and leisure scenes with 18 distinct scenes. As for natural scenes, beach, ocean, courtyard, forest, farm, pasture, snow, and desert are defined. Street, sidewalk, airfield, heliport, harbor, and railroad track are identified for transportation. Regarding sports and leisure, baseball field, basketball field, football field, and park are determined. If the scene is indoor, the mask size of each detected and segmented object in the input sketch is updated based on the following equations (1)

$$factor = \ magnifyFactor * (p_{object}/p_{total}) \quad (1)$$

, where p_{object} is the pixels' number in the object area, and p_{total} is the total pixels' number in the input sketch. $magnifyFactor$ is varied in range of (2,5) based on the determined indoor scene.

Next, for each modified object mask in indoor scenes, the mask is pasted in an empty image with the same size as the input sketch in a location determined by the following.

$$Point = ((S_{s.x} - S_{m.x})/2\,, (S_{s.y} - S_{m.y})/2) \quad (2)$$

, where $S_{s.x}$ and $S_{s.y}$ are the sketch size in x and y direction, respectively. $S_{m.x}$ and $S_{m.y}$ are the segmented object size after modifying its size in x and y direction, respectively.

Otherwise, if the scene is outdoor, no change is applied to the mask size of the segmented objects.

B. Semantic Segmentation Stage

The semantic segmentation process occurs in two levels (background and foreground segmentation). Background segmentation is first applied, where every pixel value in the background is modified based on the defined background scene in the previous stage. For each scene, we define two regions to be segmented. One exception is the beach scene, where the background is divided into three regions which are sky, sea, and sand. Following this, the pixels in each region are manipulated based on the specified region label that follows the COCO-Stuff labels. For example, in 'grass' region label, every pixel is manipulated to '124' to semantically segment the region.

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2024.3375610

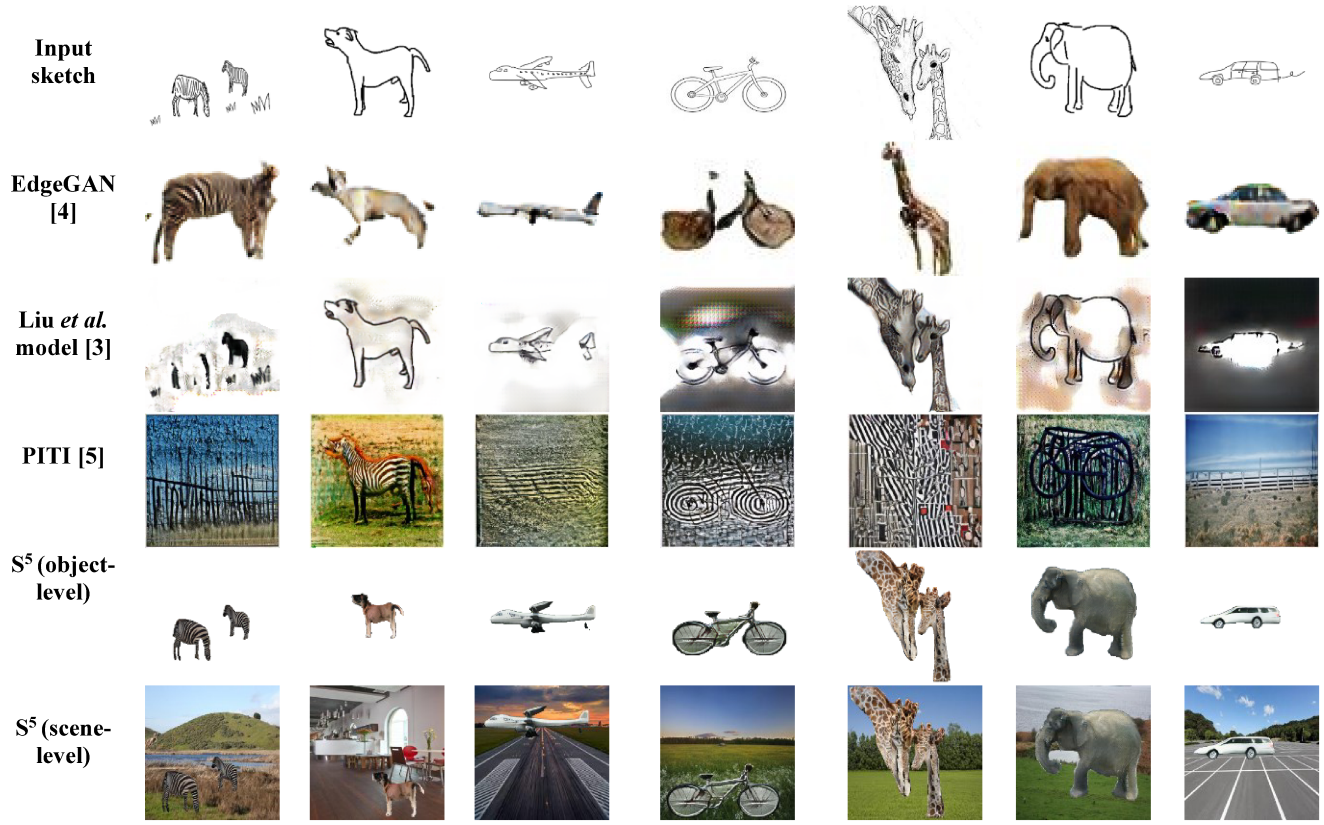


Fig. 4. A visual representation of the images produced by our approach and the baselines [3-5]. Our method maintains the objects' key structural information while generating the objects' texture. Thus, it generates better object-level results in terms of realism, quality, and fidelity. Moreover, with scene-level generated images, photorealistic and relevant backgrounds are obtained.

To make all instances that belong to a specific class easily recognizable, the foreground segmentation process is done by modifying the pixels belonging to each object in each COCO class with a particular value based on the COCO labels. Thus, foregrounds are segmented semantically and added sequentially to the generated segmented background.

Therefore, based on the first two stages, mask maps are generated as intermediate outputs. A general overview of the first and second stages is demonstrated in Figure 2.

C. *Semantic Mask-to-Image Synthesis Stage*

Colored images are generated through image-to-image translation model, in particular, CC-FPSE [13] is utilized. Image-to-image translation model takes the intermediate results, *i.e.*, semantic mask maps, produced in the first two stages as inputs and generates the texture, the color, the saturation among other features to generate colored images.

D. *Post-processing Stage*

The post-processing stage is implemented to further improve the generated images. Details are provided below.

- **Background Improvement Step.** In this step, we first classify the scene from the generated images using Places365-CNN. Then, based on the classified scene, a new scene is randomly selected from our newly collected background scene dataset. Our dataset consists of 365 different classes of scenes similar to the classes provided by Places365. However, our dataset only contains the scenes without any foreground objects to preserve the synthetic images. It contains approximately 10k images in total. These scene images are collected through Google Image search. Next, foreground objects in the generated image from previous stage are extracted using the instance and semantic segmentation stages with one difference. Only the foregrounds are segmented while ignoring the background segmentation. This mask map serves as an extractor tool to extract only the foregrounds from the generated images. After extracting the foregrounds, an alpha blending process is leveraged to blend the selected scene image with the extracted foregrounds in a specific location. This location is manually and previously determined, where each scene class is considered a cluster. Then, for each cluster, a snapping point from the background images is located so that objects are in a proper location to maintain realism.

- **Human Face Refinement Step.** It starts by detecting and localizing any human faces in the generated images after the background improvement step using a face detection model. Next, each face region is extracted, and a binarized image is formed with same size as the extracted face region. A square mask is constructed in the center of the binarized image to cover a portion of the extracted face region to complete the covered region and reconstruct the face again through an autoencoder network, namely, ICT image completion model [14]. After face reconstruction, a circular mask is applied to a

Table 1. A comparison between our method and the baselines in terms of the realism criteria, in particular, IS [19], FID [17], and LPIPS [18] on our newly collected dataset, PACS [15], TU-Berlin [16], and Sketchy dataset [2].

Method	Our newly collected dataset	PACS dataset [15]	TU-Berlin dataset [16]	Sketchy dataset [2]		
	IS [19] ↑	IS [19] ↑	IS [19] ↑	IS [19] ↑	FID [17] ↓	LPIPS [18] ↑
EdgeGAN [4]	5.49 ± 0.40	4.52 ± 0.21	5.50 ± 0.45	5.68 ± 0.27	214.806	0.49
Liu *et. al* [3]	5.18 ± 0.53	4.40 ± 0.17	4.49 ± 0.28	4.35 ± 0.15	324.082	0.66
PITI [5]	6.44 ± 0.89	6.01 ± 0.26	6.50 ± 0.56	5.93 ± 0.34	282.357	0.70
S^5 (object-level)	7.61 ± 0.45	8.43 ± 0.50	6.92 ± 0.76	7.78 ± 0.80	209.638	**0.93**
S^5 (scene-level)	**8.02 ± 0.60**	**9.82 ± 0.53**	**9.46 ± 0.50**	**11.04 ± 0.90**	**189.598**	0.76

Table 2. A comparison between our method and the baselines in terms of the fidelity criteria, in particular, L2 Distance, and SSIM [20] on our newly collected dataset, PACS [15], TU-Berlin [16], and Sketchy dataset [2].

Method	Our newly collected dataset		PACS dataset [15]		TU-Berlin dataset [16]		Sketchy dataset [2]	
	L2 Distance ↓	SSIM [20] ↑	L2 Distance ↓	SSIM [20] ↑	L2 Distance ↓	SSIM [20] ↑	L2 Distance ↓	SSIM [20] ↑
EdgeGAN [4]	1.66	0.557	1.66	0.61	1.68	0.63	1.63	0.41
Liu *et. al* [3]	1.64	0.643	1.66	0.69	1.68	0.56	1.72	0.44
PITI [5]	1.56	0.241	1.60	0.24	1.67	0.34	1.62	0.39
S^5 (object-level)	**1.44**	**0.699**	**1.58**	**0.74**	**1.66**	**0.78**	**1.52**	**0.80**

binarized image that is the same size as the reconstructed face. Then, the extracted face's size is used to resize both the reconstructed face and its respective mask. The reconstructed face is then positioned in the same exact location and blended onto the generated image. Figure 3 shows the face reconstruction and refinement step.

IV. Experiments

A. Experimental Settings

Dataset. To validate our proposed approach with prior methods, four datasets are adopted. The first dataset is our newly collected dataset that contains various sketch types since people tend to sketch differently. This dataset is acquired based on Sketchy dataset [2], ScketchyCOCO dataset [4], and through Google Image. 378 different sketches are collected containing fourteen different classes which are cat, dog, horse, sheep, cow, elephant, zebra, giraffe, car, bicycle, motorcycle, airplane, traffic light, and fire hydrant. These fourteen classes are chosen since they belong to both adopted datasets (Sketchy dataset [2], and ScketchyCOCO dataset [4]) as well as COCO dataset.

The second testing dataset is a subset of PACS dataset [15]. We choose four classes: dog, elephant, giraffe, and horse to be in our testing subset since they belong to the pre-trained state-of-the-art sketch-to-image methods and our proposed method. In total 3,081 sketches are included.

Moreover, we test our framework on a subset of TU-Berlin dataset [16]. Twenty classes are involved for the same aforementioned reason. These classes are airplane, bicycle, bus, car(sedan), cat, cow, dog, elephant, fire hydrant, giraffe, horse, motorbike, pickup truck, race car, sheep, suv, traffic light, truck, van, and zebra. The total number of images is 1600 sketches.

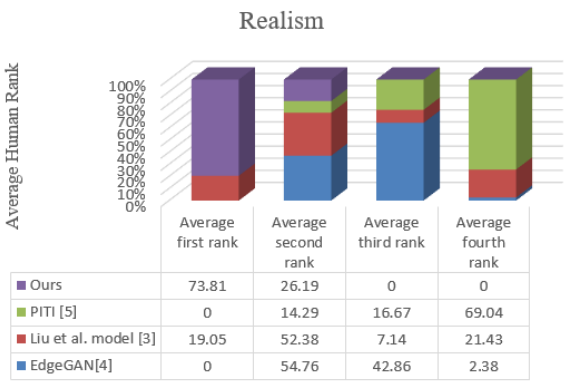
The last dataset used is a subset of Sketchy dataset [2]. It is composed of the same fourteen classes as the first evaluation dataset. In total, 1,127 sketches are included. The reason for incorporating this subset is that the ground truth images are also included.

Baselines. We compare our work with state-of-the-art methods that provide the source code. Three models are adopted to evaluate our method quantitatively and qualitatively. These models are EdgeGAN [4], the model proposed by Liu *et. al* [3], and PITI [5]. Regarding EdgeGAN [4] and PITI [5], we use the pre-trained models trained on SketchyCOCO [4] and COCO-stuff datasets, respectively. As for [3], we train the model on SketchyCOCO dataset [4] after leveraging [6] to produce pencil sketches. Since SketchyCOCO dataset [4] consists of fourteen classes, fourteen separate models are trained. This model works on generating images from sketches in two levels, where it first generates grayscale images from sketches, and then, the grayscale images are translated into colored images. Therefore, we train each model for 400 and 200 epochs for shape translation network and content enrichment network, respectively, as suggested by the authors in the original work.

B. Experimental Analysis

The objective of our research is to address and overcome the disadvantages of prior works. Previous works concentrate on generating images directly in an end-to-end manner, leading to generating unsatisfactory images, especially with complex sketches. Therefore, the objective is to decompose the sketch-to-image problem into two sub-problems to generate better results. In particular, we divided the sketch-based image generation problem into sketch-based semantic mask map and image-to-image synthesis problems. Since the GAN model is widely used in image synthesis, we integrated one of the GAN models as an image-to-image synthesis model. Then, we compared our proposed method that uses GAN with three different models. One of the baselines, particularly, PITI [5], is built based on the recently used model in image synthesis, which is the diffusion model. The experimental results validate our objective and outperform even advanced models.

(a) Fidelity criterion HR results.



(b) Realism criterion HR results

Fig. 5. The results of our proposed method and the baselines [3-5] based on the Average Human Rank (HR) of the user ranking in terms of two criteria (fidelity and realism).

Quantitative Results. To quantitatively compare our approach with the baselines, various evaluation metrics are leveraged. In particular, to evaluate the realism, FID [17] and LPIPS [18] are used. In addition, IS [19] score is computed to evaluate not only the realism but also quality and diversity. Meanwhile, to validate the faithfulness of the generated images, two different techniques are utilized. The first approach is generating the edge maps of the synthetic images based on Gabor features. Then, L2 distance is computed between the produced edge maps and the corresponding input sketches. The other technique starts by using Canny edge detector to obtain the edge maps of synthesized images. Following this, SSIM [20] is computed.

In order to compare the outputs fairly with the previous works in terms of realism criteria, object-level as well as scene-level of the generated images by our proposed framework are included. The object-level images are obtained through excluding the backgrounds from all synthetic images before background improvement step. Table 1 summarizes the comparison between our method and the baselines in terms of the realism criteria. As seen, our proposed framework outperforms the baselines. This could be attributed to two reasons. The first reason is that our framework decomposes the image generation problem into two sub-problems. This leads to first emphasize on the objects' shape during generating the mask maps, followed by concentrating on fine-grained details and rich information during generating the final outputs, *i.e.*, colored images. Thus, it produces more realistic images. The second reason is incorporating the background improvement step, where photo-realistic background scenes are blended on the generated images. This improves the images' quality as well as the images' diversity. As for the fidelity criteria, how similar the generated images are to the input sketches, our method also exceeds the baselines in L2 distance and SSIM [20], as shown in Table 2. This could be related to the decomposition of image generation problem, where the key structural information (the content) is maintained during the intermediate outputs, namely, mask maps. It is important to note that only object-level of our proposed method is evaluated in terms of the fidelity criteria for fairness reasons. The background may interfere during computing the similarity between the sketched objects and the generated objects; thus, the background is excluded. Visual comparison is illustrated in Figure 4. As shown in Figure 4, our proposed method achieves the best results in terms of quality, realism, diversity, and fidelity. Our method maintains the objects' key structural information while generating the objects' texture. Thus, it generates better object-level results. Moreover, with scene-level generated images, photorealistic and relevant backgrounds are obtained.

Qualitative Results. We conduct a perceptual study to qualitatively assess the synthetic images based on two main criteria (realism and fidelity). Our sample consists of 45 participants ages 20-50 years old with 25 females and 20 males. The participants are requested to rank the synthesized images obtained by our method and the baselines based on the realism criterion and on a scale from 1 to 4, where 1 indicates most realistic image and 4 refers to least realistic image. Additionally, they are asked to rank the synthesized images of our approach and the baselines based on the similarity between the generated images to the input sketches and the coloring quality. Same scale is used as well for fidelity criterion. Following the users' ranking, we compute the Average Human Rank (HR) depending on the user ranking choices. HR for both realism and fidelity criteria are shown in Figure 5. As seen, our proposed framework significantly defeats the baselines in both criteria.

V. Conclusion and Future Work

Nowadays image generation is becoming a trending topic. State-of-the-art sketch-based image generation methods have shown great potential; however, the results for complicated sketches are still not satisfactory. To this end, this paper proposes S^5, a novel sketch-to-image synthesis framework, where intermediate outputs, namely, semantic mask maps are first generated from the input sketches through instance and semantic segmentation. Our model not only concentrates on the sketched objects, but it is also aware of the surrounding environment to reflect reality and enhance the generation process. Then, these mask maps are translated into colored images. This way, the structural information, the shape, the orientation, the occlusion, among other features are maintained since sketches are not directly translated into images in one-

shot. Following this, we further improve our generated images incorporating two additional steps which are background improvement and face refinement steps.

Our approach represents a substantial advancement in image generation, where it surpasses the previous work in sketch-to-image synthesis field in terms of realism, quality, diversity, and fidelity. Various collections of evaluation datasets have been used during the evaluation process. Indeed, our method is able to generate images of complex sketches as well as to produce images from different sketch styles. For future work, we plan to integrate more advanced components, such as the diffusion model in the semantic mask-to-image synthesis stage. We believe that the integration of more advanced components will produce better results. Furthermore, we aim to roughly estimate and approximate the input sketches to the closest edge maps to further enhance our results. This would enable the creation of images even from highly abstract sketches.

ACKNOWLEDGMENT

The first author would like to thank Umm Al-Qura University, in Saudi Arabia, for the continuous support. The second author is supported by NSF grant # 2025234. This work has been supported in part by the University of Dayton Office for Graduate Academic Affairs through the Graduate Student Summer Fellowship Program.

REFERENCES

[1] G. G. Rajput and Prashantha, “Sketch based image retrieval using grid approach on large scale database,” Procedia Comput. Sci., vol. 165, pp. 216–223, 2019.

[2] P. Sangkloy, N. Burnell, C. Ham, and J. Hays, “The sketchy database: Learning to retrieve badly drawn bunnies,” ACM Trans. Graph., vol. 35, no. 4, pp. 1–12, 2016.

[3] R. Liu, Q. Yu, and S. X. Yu, “Unsupervised sketch to photo synthesis,” in Computer Vision – ECCV 2020, Cham: Springer International Publishing, 2020, pp. 36–52.

[4] C. Gao, Q. Liu, Q. Xu, L. Wang, J. Liu, and C. Zou, “SketchyCOCO: Image generation from freehand scene sketches,” in 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2020, pp. 5173–5182.

[5] T. Wang, T. Zhang, B. Zhang, H. Ouyang, D. Chen, Q. Chen, F. Wen, “Pretraining is all you need for image-to-image translation,” arXiv [cs.CV], 2022.

[6] M. Beyeler, "OpenCV with Python Blueprints: Design and develop advanced computer vision projects using OpenCV with Python", Packt Publishing Ltd., London, England, ISBN 978-178528269-0, 2015.

[7] S. Xie and Z. Tu, “Holistically-nested edge detection”, Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2015.

[8] S. Qiao, L.-C. Chen, and A. Yuille, “DetectoRS: Detecting objects with recursive feature pyramid and switchable atrous convolution,” in 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021.

[9] C. Zhang, Y. Tang, C. Zhao, Q. Sun, Z. Ye, and J. Kurths, “Multitask GANs for semantic segmentation and depth completion with cycle consistency,” IEEE Trans. Neural Netw. Learn. Syst., vol. 32, no. 12, pp. 5404–5415, 2021.

[10] Gao, J., Wang, Q., and Yuan, Y, “Feature-aware Adaptation and Structured Density Alignment for Crowd Counting in Video Surveillance,” ArXiv, abs/1912.03672. 2019.

[11] T. Han, J. Gao, Y. Yuan, and Q. Wang, “Unsupervised semantic aggregation and deformable template matching for semi-supervised learning,” in Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020, pp. 9972–9982.

[12] B. Yang , F. Wan , C. Liu , B. Li , X. Ji , and Q. Ye, "Part-Based Semantic Transform for Few-Shot Semantic Segmentation," in IEEE Transactions on Neural Networks and Learning Systems, 2021, DOI:10.1109/tnnls.2021.3084252.

[13] X. Liu, G. Yin, J. Shao, X. Wang, and H. Li, “Learning to predict layout-to-image conditional convolutions for semantic image synthesis,” arXiv [cs.CV], 2019.

[14] Z. Wan, J. Zhang, D. Chen, and J. Liao, “High-fidelity pluralistic image completion with trans-formers,” arXiv [cs.CV], 2021.

[15] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, “Deeper, broader and artier domain generalization,” in 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 5542–5550.

[16] M. Eitz, J. Hays, and M. Alexa, “How do humans sketch objects?” ACM Trans. Graph., vol. 31, no. 4, pp. 1–10, 2012.

[17] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," In NIPS, 2017.

[18] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[19] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, “Improved techniques for training GANs,” in Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 2234–2242.

[20] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” IEEE Trans. Image Process., vol. 13, no. 4, pp. 600–612, 2004.