ELSEVIER

Pattern Recognition Letters
journal homepage: www.elsevier.com

Improving Sketch Colorization using Adversarial Segmentation Consistency

Samet Hicsonmez[a,**], Nermin Samet[b], Emre Akbas[c], Pinar Duygulu[a]

[a] *Hacettepe University, Ankara, Turkey*
[b] *LIGM, Ecole des Ponts, Univ Gustave Eiffel, CNRS, Marne-la-Vallée, France*
[c] *Middle East Technical University, Ankara, Turkey*

arXiv:2301.08590v1 [cs.CV] 20 Jan 2023

Article history:

sketch colorization, sketch to image translation, Generative Adversarial Networks (GAN), image segmentation, image to image translation

ABSTRACT

We propose a new method for producing color images from sketches. Current solutions in sketch colorization either necessitate additional user instruction or are restricted to the "paired" translation strategy. We leverage semantic image segmentation from a general-purpose panoptic segmentation network to generate an additional adversarial loss function. The proposed loss function is compatible with any GAN model. Our method is not restricted to datasets with segmentation labels and can be applied to unpaired translation tasks as well. Using qualitative, and quantitative analysis, and based on a user study, we demonstrate the efficacy of our method on four distinct image datasets. On the FID metric, our model improves the baseline by up to 35 points. Our code, pretrained models, scripts to produce newly introduced datasets and corresponding sketch images are available at https://github.com/giddyyupp/AdvSegLoss.

© 2023 Elsevier Ltd. All rights reserved.

1. Introduction

The task of image generation from an input sketch or edge map is known as "sketch to image translation", or "sketch colorization". Sketches capture essential content of the images and they can be easily acquired. Yet, vast amount of domain difference between single channel edge maps and color images makes sketch colorization a challenging process. Lack of details in sketches especially for background is another problem.

Sketch colorization has been investigated in numerous domains: faces [1, 2, 3, 4], objects [5, 6, 7, 8], animes [9, 10, 11, 12, 13, 14, 15], art [16], icons [17] and scenes [18, 19, 20]. The majority of the approaches require user direction in the form of supplementary input, such as a reference color, patch, or image. These approaches usually generate surreal colorizations otherwise. Except for a few studies (e.g., Liu et al. [8]), most of the approaches follow the “paired” strategy, which is restricted to use datasets with a ground-truth image for each sketch.

In this study, we aim to leverage general purpose semantic image segmentation to alleviate the aforementioned shortcomings. We argue that an accurately colored sketch would produce a “real” segmentation result, i.e., a result that looks like the segmentation of a real image. Thus, for sketch based image colorization problem, we exploit semantic segmentation methods that have reached to a degree of maturity even for datasets on which they were not trained (Section 4). We introduce a segmentation-based adversarial loss to be used in a GAN (Generative Adversarial Network) setup. With our approach, neither extra user instruction nor "paired" input is required.

We introduce three models for varying levels of segmentation feedback in the sketch to image translation pipeline. Our models could be integrated into both paired and unpaired GAN models. We illustrate the effectiveness of applying segmentation cues via comprehensive experimental analyses. This paper extends our previous work [21] in the following ways: (i) We apply our method to a new task: label-to-photo translation. Our experiments on two challenging datasets show that our segmentation-based adversarial loss is useful in this task, too. Again, ground-truth segmentation labels are not a requirement for our approach. (ii) We perform experiments to set the optimal values for weights of two additional discriminators. (iii) We incorporate an outdoor dataset, Cityscapes [22], to both

[**] Corresponding author
e-mail: samethicsonmez@hacettepe.edu.tr (Samet Hicsonmez), nermin.samet@enpc.fr (Nermin Samet), emre@ceng.metu.edu.tr (Emre Akbas), pinar@cs.hacettepe.edu.tr (Pinar Duygulu)

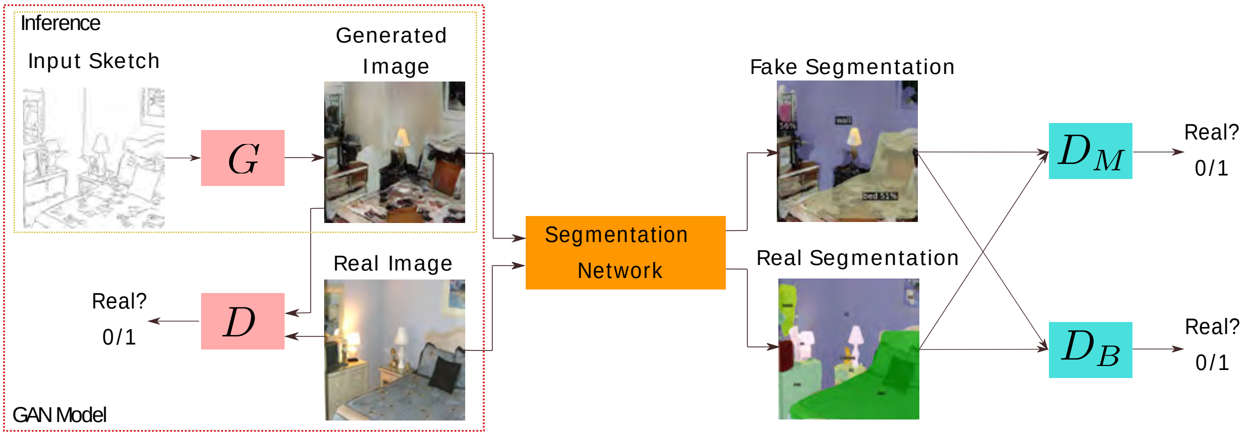

Fig. 1: The proposed model for sketch colorization with Adversarial Segmentation Loss (ASL). It is composed of two parts; a general purpose image translation GAN model, and an image segmentation model. During training, input sketches are first colorized using the baseline GAN model. Then, generated and ground truth color images are fed to the pre-trained panoptic segmentation model to extract fake and real segmentation maps. Finally, two additional discriminators are used to classify the segmentation maps as fake or real, respectively. The box with dashed yellow borders shows the inference stage. Red border marks the GAN model used for sketch to image translation. Here, Pix2Pix is used as an example image translation model, which could be replaced by any paired or unpaired model.

sketch colorization and label to photo translation tasks. (iv) We add a new metric, mean Intersection over Union (mIoU), in addition to FID score to measure the performance of all the models more reliably.

2. Related Work

Even though the sketch and the edge map of an image are different concepts, in practice, XDoG [23] or HED [24] based edge maps are considered as sketches (e.g., [18, 11]). Moreover, some sketch based models [5] use edge maps for data augmentation. Hence, we refer to all these models as "sketch-to-image translation" or "sketch colorization" models. Although general purpose image-to-image translation methods [25, 26, 27, 28, 29] could be used for sketch-to-image translation tasks, the results are not satisfactory.

One widely used solution to improve the colorization performance is to employ additional color [18, 11, 12, 10, 14, 15], patch [6], image [2, 13, 16, 9, 17, 30] or language guidance [20, 31, 32]. For instance, in color guidance, users specify their desired colors for the regions in the sketch image, and the model utilizes this information to generate the same or similar colors for these regions. Some automatic methods also utilize user guidance to improve their performance as a hybrid approach. Most of the sketch-to-image translation methods are based on “paired” training approach [18, 11, 5, 19], but, recently unpaired methods have also been presented [8, 16].

Scribbler [18] presents one of the very first paired and user guided scene sketch colorization models. In addition to pixel, perceptual and GAN losses, Scribbler uses total variation loss to encourage smoothness. XDoG is used to generate sketch images of 200k bedroom photos. DCSGAN [15] uses HSV color space in addition to the RGB, for line art colorization task. Zou et al. [20] use text inputs to progressively colorize an input sketch, in such a way that a novel text guided sketch segmenter locates the objects in the scene. EdgeGAN [19] maps edge images to a latent space during training using an edge encoder. During inference, the edge encoder is used to encode the input sketch to the latent space to subsequently generate a color image. Experiments are provided for 14 foreground and 3 background objects from COCO [33] dataset.

EdgeGAN [19] and Scribbler [18] use a supervised approach where input sketches and corresponding output images exist. However, it is hard to collect sketch image pairs. Liu et al. [8] propose a two stage method to convert object sketches to color images in an unsupervised (unpaired) way. They first convert sketches to gray scale images, and then to color images. Self supervision is used to complete the deliberately deleted sketch parts and clear the added noisy edges from sketch images.

In Sketch-to-Art [16], an art image is generated using an input sketch, with the additional help of the target style art image. Content of the input sketch and style of the art image are encoded, and then fused to generate a stylized art image. In [17] authors proposed a method to colorize icons which utilizes colored icon images as input in addition to the black-white icons.

The user input is valuable not only in helping the colorization network to put the right colors to indicated regions, but also in removing the color bleeding problems. In [34], users draw scribbles to the regions on the generated image suffered from color bleeding artifacts for guiding the model to fix them.

Unlike these methods, our method does not require any user input to generate satisfactory colorization. Instead, we utilize adversarial segmentation guidance to improve performance.

3. Adversarial Segmentation Consistency

Figure 1 shows the overall structure of our proposed model for sketch colorization, which we refer to as Adversarial Segmentation Loss (ASL) based model. In this work, we used *Pix2Pix* and *CycleGAN* methods as our baselines for paired and unpaired training, respectively. This preference is made based on the effectiveness of these methods across a variety of tasks and datasets. In the figure, Pix2Pix is used to show ASL based model for paired approach. Our model could be integrated into any other paired or unpaired GAN model.

Our model consists of a baseline GAN, a panoptic segmentation network (*Seg*) and two discriminators (D_M and D_B). Panoptic segmentation network is trained offline on the COCO Stuff [35] dataset and its weights are frozen during the training of our model. Fake and real images are fed to the *Seg* network to get real and fake segmentation maps. Then, these two segmentation maps are given to the discriminators to classify them as fake or real. We designed three variants of our model to embed different levels of segmentation feedback to the sketch to image translation pipeline.

The first variant utilizes the full (multiclass) segmentation map of an image where all foreground and background classes (a total of 135 classes) are considered. In this model, ground-truth color image I_{real} and the generated color image I_{fake} are fed to *Seg* which outputs full segmentation maps for both images. Then, these two outputs are given to a discriminator network D_M to discriminate between real and fake segmentation maps. We call this model as *Multi-class* in the rest of the paper.

As a higher level of abstraction, grouping objects as background and foreground alone may yield sufficient information. The second variant of our model uses only two classes (background and foreground) in the segmentation map by grouping all foreground classes into one and all background classes into another class. In this model, which we refer to as *Binary*, binary segmentation outputs for real and fake images are fed to a discriminator network D_B to discriminate between real and fake ones. Finally, our third variant is the union of the above two. It contains both discriminators, and is named as *Combined*.

Overall loss function for our model is the sum of losses of the baseline GAN model (L_G) and the two additional discriminators' (L_B and L_M). That is, the objective function is:

$$\mathcal{L} = w_g L_G + w_b L_B + w_m L_M$$

Let Seg_B and Seg_M correspond to the panoptic segmentation networks in *Binary* and *Multi-class* cases. The additional losses that we introduce, L_B and L_M, are defined as:

$$\begin{aligned}\mathcal{L}_B(G, D_B, Seg_B) = & \sum_i log(D_B(Seg_B(y_i)))+ \\ & \sum_i log(1 - D_B(Seg_B(G(x_i)))\end{aligned}$$

$$\begin{aligned}\mathcal{L}_M(G, D_M, Seg_M) = & \sum_i log(D_M(Seg_M(y_i)))+ \\ & \sum_i log(1 - D_M(Seg_M(G(x_i)))\end{aligned}$$

Let x_i be an input sketch image, and y_i be the corresponding ground truth color image. When the baseline GAN model is Pix2Pix [27], GAN loss L_G is formulated as:

$$\mathcal{L}_G(G, D) = \sum_i log(D(x_i)) + \sum_i log(1 - D(G(x_i)) + \sum_i \|y_i - G(x_i)\|$$

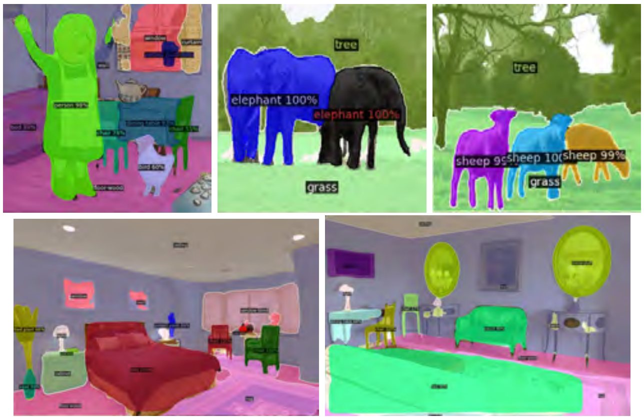

Fig. 2: Sample segmentations using general purpose panoptic segmentation network on different datasets. The model generalizes well to several domains.

Table 1: Statistics of the datasets used in our experiments.

Dataset	Train Images	Test Images
ADE20k Bedroom	1355	135
Cityscapes	2975	500
Illustration	659	131
COCO Elephant	1800	343
COCO Sheep	1300	229

When baseline is CycleGAN [25], L_G for direction $X \rightarrow Y$ is:

$$\begin{aligned}\mathcal{L}_G(G_X, G_Y, D_X) = & \sum_j log(D_X(y_j)) + \sum_i log(1 - D_X(G_X(x_i)) \\ & + \sum_i \|x_i - G_Y(G_X(x_i))\|\end{aligned}$$

where G_X maps input sketches to color images, and G_Y maps the color images back to sketch domain. D_X is the discriminator for domain X, i.e. sketches. Final L_G for CycleGAN is the sum of above formulation for two directions.

We analysed the effect of each component in the objective function and, set w_g, w_b and w_m to 1 based on the experimental analysis (see Section 5.1). Note that, for the *Binary* model w_m, and for the *Multi-class* model w_b is set to 0 respectively.

4. Datasets

We evaluated our models on five challenging datasets (see Table 1). The first dataset consists of bedroom images from the ADE20k indoor dataset [36], with 1355 train and 135 test images. The second dataset is Cityscapes [22] dataset which contains 2975 training and 500 test images. The third dataset [28] contains illustrations from children's books by Alex Scheffler, with 659 train and 131 test images. The fourth and fifth ones were curated by us from the COCO dataset. We collected images containing elephant or sheep. Note that these images may also contain other foreground/background objects such as person, animals, mountains, grass and sky. Elephant dataset contains 1800 train and 343 test images, and the sheep dataset has 1300 train and 229 test images. Example images from these datasets and their segmentation outputs are shown in Figure 2.

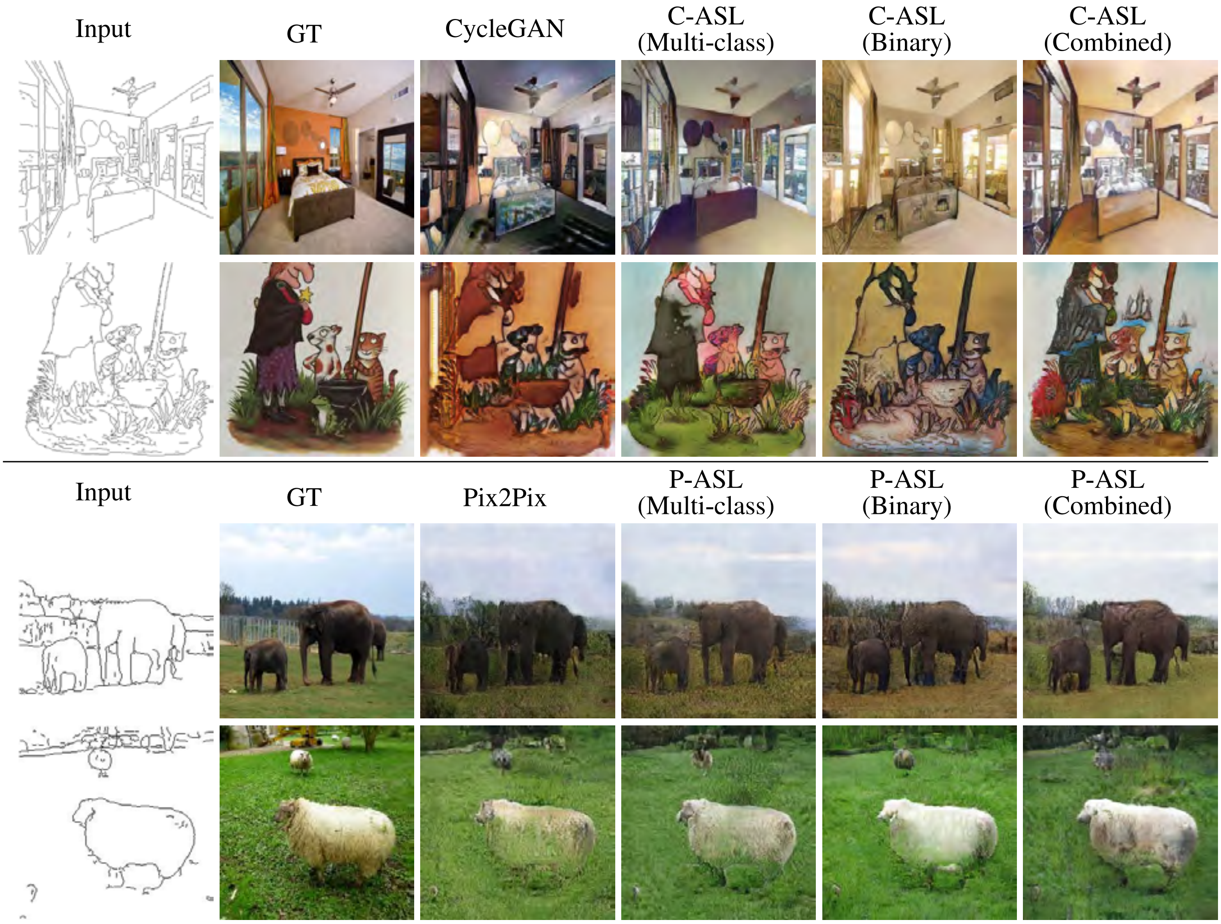

Fig. 3: Sample results from baselines and our model with different settings. Input images on each row are from bedroom, illustration, elephants and sheep datasets, respectively. First two rows display results of unpaired training (baseline is CycleGAN), and last two rows show results for paired training (baseline is Pix2Pix). On bedroom and elephant datasets *Binary*, on illustration and sheep datasets *Combined* setting gave best results for both training schemes.

Table 2: Comparison with CycleGAN [25] on **unpaired sketch-to-image translation** task in terms of **FID scores**, lower is better.

Dataset	CycleGAN	C-ASL (Multi-class)	C-ASL (Binary)	C-ASL (Combined)
Bedroom	113.1	111.7	**87.1**	93.2
Cityscapes	62.9	64.1	64.9	**59.1**
Illustration	213.6	206.9	204.8	**189.4**
Elephant	126.4	103.9	**91.9**	116.9
Sheep	209.3	207.2	236.1	**196.8**

Edge images are extracted using the HED [24] method. In the first two columns of Figure 3, we present sample natural and edge images for all the datasets. It can be seen that the images contain a variety of foreground and background objects, also it is hard to figure out the source dataset for some images.

5. Experiments

We used PyTorch [37] to implement our models. We use sketch images as source domain, and color images as target domain. All training images (i.e. color and sketch images) are resized to 256×256 pixels. We train all models for 200 epochs using the Adam optimizer [38] with a learning rate of 0.0002. We conducted all our experiments on a NVIDIA Tesla V100 GPU.

We compared our models with Pix2Pix [27] and AutoPainter (AP) [11] for paired and CycleGAN [25] for unpaired setting on **sketch-to-image translation** task. We used the official implementations that are publicly available. Baseline models are trained for 200 epochs. Our proposed ASL model that uses Pix2Pix as the baseline GAN model is referred to as P-ASL, and similarly C-ASL refers to the model that uses CycleGAN.

5.1. Quantitative Analysis

To quantitatively evaluate the quality of generated images, we used the widely adopted Frechet Inception Distance (FID) [39] metric. FID score measures the distance between the distributions of the generated and real images. Lower FID score indicates the higher similarity between two image sets.

On Bedroom and Cityscapes datasets where ground truth segmentation maps are available, we also calculate the mean Intersection over Union (mIoU) scores on colorized images. We forward each colorized image to an off-the-shelf segmentation model trained on these two datasets separately. mIoU score measures the quality of the segmentation. We argue that better colorized images should yield higher mIoU scores.

Table 3: Comparison with AutoPainter [11] and Pix2Pix [27] on **paired sketch-to-image translation** task in terms of **FID scores**, lower is better.

Dataset	Auto Painter	Pix2Pix	P-ASL (Multi-class)	P-ASL (Binary)	P-ASL (Combined)
Bedroom	206.8	100.5	100.0	**95.1**	110.1
Cityscapes	151.3	74.1	**69.9**	71.6	71.2
Illustration	272.0	180.0	176.9	178.0	**175.7**
Elephant	155.1	83.5	85.8	**78.8**	82.8
Sheep	233.1	157.0	159.9	162.0	**150.5**

Table 4: Comparison with CycleGAN on **unpaired sketch-to-image translation** task in terms of **mIoU scores**, higher is better.

Dataset	CycleGAN	C-ASL (Multi-class)	C-ASL (Binary)	C-ASL (Combined)	Oracle
Bedroom	5.20	**6.71**	6.58	6.44	20.62
Cityscapes	15.67	14.63	13.56	**15.68**	44.85

We present FID scores for unpaired translation in Table 2 and paired translation in Table 3. FID scores are inline with the visual inspections (see Figure 3), for all the datasets, at least one variant of our model performed better than the baseline.

First of all, when we compare FID scores of two training schemes and baseline models, paired training (Pix2Pix) performed better than unpaired training, as expected. However, our "adversarial segmentation loss" affected the results of paired and unpaired cases differently. For instance, on elephant dataset our models improved baseline up to 35 points for unpaired case, but only 5 points for paired case.

Another crucial observation is that segmentation guidance closed the gap between unpaired and paired training results. Best FID scores for unpaired models on bedroom, illustration and elephant datasets become very close to or even better than paired training. For instance on the elephant dataset, the initial 40+ point FID gap (126 vs 83) dropped to 13 (92 vs 79) on *Binary* setting. Here the only exception is the sheep dataset. Since the sheep dataset contains various complex objects, unpaired and paired models failed to generate plausible images.

We show mIoU scores for unpaired translation in Table 4 and paired translation in Table 5. We also present oracle performances of the segmentation method on both datasets. On mIoU metric, again for all the datasets, at least one of the variants of our model performed better than the baseline.

When we look at the best performing settings on different datasets, structure of the dataset has an effect on the results. For instance, even though one is an indoor and the other one is an outdoor dataset, bedroom and elephant images are composed of similar structure. FG/BG ratios and placements of them in these datasets are similar across all images, i.e. walls, ceiling and floors in bedroom images are always positioned in the same places on different images. Also elephant images contain very few FG objects, i.e. only elephants most of the time, and large BG areas such as grass, trees and sky. On these two datasets, *Binary* setting which considers FG/BG classes only gave the best FID score. On the other hand, illustration and sheep images got a variety of FG objects and scenes. On such datasets, using only a FG/BG discriminator even degrades the performance.

Table 5: Comparison with Pix2Pix on **paired sketch-to-image translation** task in terms of **mIoU scores**, higher is better.

Dataset	Auto Painter	Pix2Pix	P-ASL (Multi-class)	P-ASL (Binary)	P-ASL (Combined)	Oracle
Bedroom	2.08	6.49	6.95	**7.39**	6.60	20.62
Cityscapes	6.02	18.71	18.63	18.70	**18.74**	44.85

Table 6: User Study results.

Dataset	CycleGAN	C-ASL
Bedroom	20.0	**80.0**
Illustration	27.0	**73.0**
Elephant	39.1	**60.9**
Sheep	19.1	**80.9**
Cityscapes (L2P)	25.7	**74.3**

Table 7: Effect of changing the w_b and w_m values, w_b is used as 1.0 for all experiments. Using 1.0 for both weights yields the best FID score on the ADE20k bedroom images for the task of sketch-to-image translation.

	w_b and w_m				
	0.1	0.5	1.0	5.0	10.0
FID	114.8	114.5	**93.2**	147.8	104.6

Our model has two important parameters, w_b and w_m, to control the effect of segmentation discriminators. To find the best possible values, we conducted experiments by training our models on the ADE20k bedroom images on the unpaired sketch to image translation task (see Table 7). Using a small value like 0.1 gives a similar score to baseline CycleGAN. On the other hand using a big value like 5.0 increased the FID score dramatically. Setting w_b and w_m to 1.0 resulted in the best FID score, thus the weights are set to 1.0 in all experiments.

5.2. *Qualitative Analysis and User Study*

We present visual results of sketch colorization for our model and the baseline models in the Figure 3. On bedroom and illustration datasets, we show results of unpaired training, and on elephant and sheep datasets we show paired training results.

On the bedroom dataset, the *Binary* setting generates better images compared to baselines and other settings. Colors are uniform across the object parts in this setting. There are defective colors in the CycleGAN results such as the bottom of the bed and floor. On the illustration dataset, the baseline model performed poorly. Objects are hard to recognize and most importantly colors are not proper at all. On the other hand, *Multi-class* and *Combined* settings generate significantly better images i.e. generated objects and background got consistent colors. Finally, on elephant and sheep datasets although generated images are not very visually appealing for all the methods, segmentation guided images are quite appealing compared to baseline models'. On the elephant dataset *Binary*, on the sheep dataset *Combined* setting performed the best.

We conducted a user study to measure realism of generated images. We show two random images (at random positions,

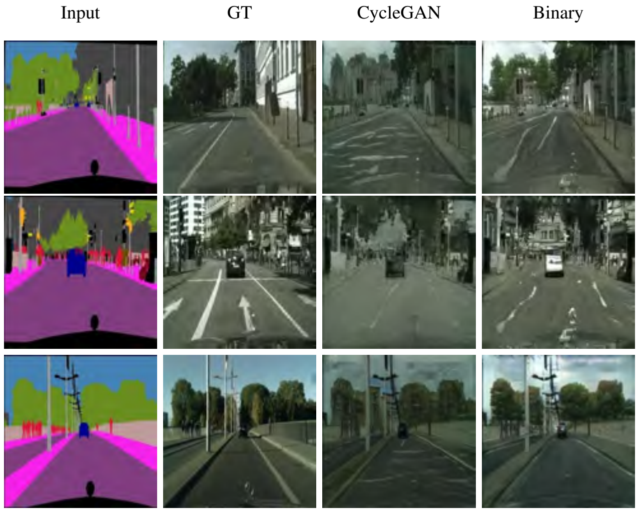

Fig. 4: Sample results on Cityscapes dataset for CycleGAN and our best model. In the first image, CycleGAN generates buildings instead of trees, also there are defects on the road. Other two images are blurry and lack details. On the other hand, in all cases our *Binary* setting generates more visually appealing images.

Table 8: Comparison with CycleGAN [25] on **unpaired label-to-photo translation** task in terms of **FID scores**, lower is better.

Dataset	CycleGAN	C-ASL (Multi-class)	C-ASL (Binary)	C-ASL (Combined)
Bedroom	84.2	86.9	85.6	**78.9**
Cityscapes	83.0	70.9	**64.8**	66.0

left or right) which were generated with CycleGAN and our best setting (lowest FID score) for all four datasets, and asked participants to select the more realistic one.

We collected a total of 115 survey inputs from 39 different users. We ask users to evaluate 4 S2P models and 1 L2P model. In Table 6, we present results of the user study in terms of preference percentages of each model. User study results are inline with the FID score results, on all datasets, images generated by our model were preferred by the users most of the time. On sheep and elephant datasets, users struggled to select an answer. Color distributions and shapes of FG objects are two dominant factors which lead user preferences.

5.3. Label to Photo Translation

We also experimented with label-to-photo (L2P) translation task to show the effectiveness of our model in a different task where adversarial segmentation loss could be helpful. In L2P task, we use ADE20k bedroom and Cityscapes datasets. Similar to S2P task, all images are resized to 256x256 pixels. We train L2P models for 200 epochs using the Adam optimizer [38] with a learning rate of 0.0002. We show FID scores for unpaired L2P in Table 8 and paired translation in Table 9. For unpaired translation our best performing method improves the baseline for more than 5 points on Bedroom and almost 20 points on Cityscapes datasets. Similarly on the paired translation, the improvements regard to the baseline reaches 18 points.

Table 9: Comparison with Pix2Pix [27] on **paired label-to-photo translation** task in terms of **FID scores**, lower is better.

Dataset	Pix2Pix	P-ASL (Multi-class)	P-ASL (Binary)	P-ASL (Combined)
Bedroom	128.1	118.2	122.3	**110.1**
Cityscapes	79.5	78.4	**72.9**	77.6

Table 10: Comparison with CycleGAN on **unpaired label-to-photo translation** task in terms of **mIoU scores**, higher is better.

Dataset	CycleGAN	C-ASL (Multi-class)	C-ASL (Binary)	C-ASL (Combined)	Oracle
Bedroom	5.70	5.88	6.10	**6.69**	20.62
Cityscapes	20.13	**21.45**	20.70	19.61	44.85

Table 11: Comparison with Pix2Pix on **paired label-to-photo translation** task in terms of **mIoU scores**, higher is better.

Dataset	Pix2Pix	P-ASL (Multi-class)	P-ASL (Binary)	P-ASL (Combined)	Oracle
Bedroom	1.56	1.59	1.54	**1.62**	20.62
Cityscapes	8.62	8.66	**8.69**	8.56	44.85

In Table 10 and Table 11, we present mIoU scores for unpaired and paired L2P translation, respectively. For both cases, our best performing variant outperforms the baseline method.

We present visual results in Figure 4 for only the baseline model and our best performing setting *Binary* for unpaired translation. Our model generates more photo-realistic images, also generated images comply with the input label maps better.

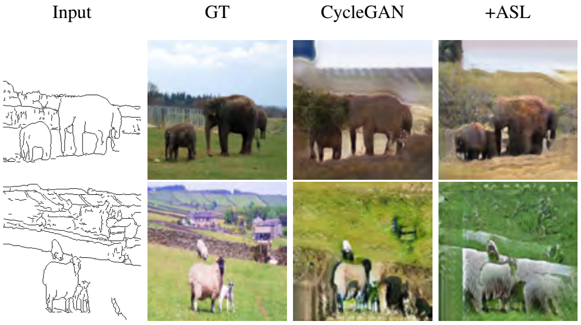

Fig. 5: Sample results on elephant and sheep datasets for CycleGAN and our best model. Realism of both models are not satisfactory, however, especially colors of BG areas are better in our results.

5.4. Limitations

Figure 5 presents examples on elephant and sheep datasets where both baseline and our best performing model suffer from low visual realism. The main reason for that is these datasets contains complex foreground and background objects. However, our method performs significantly better than the baseline. Especially on the first row, colorized image using our method resembles more to the ground truth image.

6. Conclusion

In this study, we present a new method for the sketch colorization problem. Our method utilizes a general purpose image segmentation network and adds an adversarial segmentation loss (ASL) to the regular GAN loss. ASL could be integrated to any GAN model, and works even if the dataset does not have segmentation labels. We used CycleGAN and Pix2Pix as baseline GAN models. We conducted extensive evaluations on various datasets including bedroom, sheep, elephant and illustration images and evaluate the performance both quantitatively (using FID and mIoU scores) and qualitatively (through a user study). We showed that our model outperforms baselines on all datasets on both FID score and user study analysis.

Regarding the limitations of our method, although we improve the baseline both qualitatively and quantitatively, especially elephant and sheep results lack realism. Even the paired training results are not visually appealing on these two datasets, most probably due to the fact that the baseline models are not very successful at generating complex scenes.

Acknowledgment

The numerical calculations reported in this paper were fully performed at TUBITAK ULAKBIM, High Performance and Grid Computing Center (TRUBA resources). Dr. Akbas is supported by the “Young Scientist Awards Program (BAGEP)” of Science Academy, Turkey.

References

[1] S.-Y. Chen, W. Su, L. Gao, S. Xia, H. Fu, Deepfacedrawing: deep generation of face images from sketches, ACM Transactions on Graphics (TOG) 39 (4) (2020) 72–1.

[2] J. Lee, E. Kim, Y. Lee, D. Kim, J. Chang, J. Choo, Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[3] Y. Li, X. Chen, F. Wu, Z.-J. Zha, Linestofacephoto: Face photo generation from lines with conditional self-attention generative adversarial networks, in: ACM International Conference on Multimedia, 2019.

[4] J. Huang, L. Jing, Z. Tan, S. Kwong, Multi-density sketch-to-image translation network, IEEE Transactions on Multimedia 24 (2022) 4002–4015.

[5] W. Chen, J. Hays, Sketchygan: Towards diverse and realistic sketch to image synthesis, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[6] W. Xian, P. Sangkloy, V. Agrawal, A. Raj, J. Lu, C. Fang, F. Yu, J. Hays, Texturegan: Controlling deep image synthesis with texture patches, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[7] Y. Lu, S. Wu, Y.-W. Tai, C.-K. Tang, Image generation from sketch constraint using contextual gan, in: IEEE European Conference on Computer Vision, 2018.

[8] R. Liu, Q. Yu21, S. X. Yu, Unsupervised sketch to photo synthesis, in: IEEE European Conference on Computer Vision, 2020.

[9] C. Furusawa, K. sHiroshiba, K. Ogaki, Y. Odagiri, Comicolorization: semi-automatic manga colorization, in: SIGGRAPH Asia Technical Briefs, 2017.

[10] Y. Ci, X. Ma, Z. Wang, H. Li, Z. Luo, User-guided deep anime line art colorization with conditional adversarial networks, in: ACM International Conference on Multimedia, 2018.

[11] Y. Liu, Z. Qin, T. Wan, Z. Luo, Auto-painter: Cartoon image generation from sketch by using conditional wasserstein generative adversarial networks, Neurocomputing 311 (2018) 78 – 87.

[12] L. Zhang, C. Li, T.-T. Wong, Y. Ji, C. Liu, Two-stage sketch colorization, ACM Transactions on Graphics (TOG) 37 (6) (2018) 1–14.

[13] L. Zhang, Y. Ji, X. Lin, C. Liu, Style transfer for anime sketches with enhanced residual u-net and auxiliary classifier gan, in: Asian Conference on Pattern Recognition, 2017.

[14] M. Yuan, E. Simo-Serra, Line art colorization with concatenated spatial attention, in: CVPR Workshops, 2021.

[15] Z. Dou, N. Wang, B. Li, Z. Wang, H. Li, B. Liu, Dual color space guided sketch colorization, IEEE Transactions on Image Processing 30 (2021) 7292–7304.

[16] B. Liu, K. Song, Y. Zhu, A. Elgammal, Sketch-to-art: Synthesizing stylized art images from sketches, in: Proceedings of the Asian Conference on Computer Vision, 2020.

[17] Y.-k. Li, Y.-H. Lien, Y.-S. Wang, Style-structure disentangled features and normalizing flows for diverse icon colorization, in: IEEE Conference on Computer Vision and Pattern Recognition, 2022, pp. 11244–11253.

[18] P. Sangkloy, J. Lu, C. Fang, F. Yu, J. Hays, Scribbler: Controlling deep image synthesis with sketch and color, in: IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[19] C. Gao, Q. Liu, Q. Xu, L. Wang, J. Liu, C. Zou, Sketchycoco: Image generation from freehand scene sketches, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[20] C. Zou, H. Mo, C. Gao, R. Du, H. Fu, Language-based colorization of scene sketches, ACM Transactions on Graphics (TOG) 38 (6) (2019) 1–16.

[21] S. Hicsonmez, N. Samet, E. Akbas, P. Duygulu, Adversarial segmentation loss for sketch colorization, in: IEEE International Conference on Image Processing, 2021.

[22] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: IEEE Conference on Computer Vision and Pattern Recognition, 2016.

[23] H. Winnemöller, J. E. Kyprianidis, S. C. Olsen, Xdog: an extended difference-of-gaussians compendium including advanced image stylization, Computers & Graphics 36 (6) (2012) 740–753.

[24] S. Xie, Z. Tu, Holistically-nested edge detection, in: IEEE International Conference on Computer Vision, 2015.

[25] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, Unpaired image-to-image translation using cycle-consistent adversarial networks, 2017.

[26] Z. Yi, H. Zhang, P. Tan, M. Gong, Dualgan: Unsupervised dual learning for image-to-image translation, 2017.

[27] P. Isola, J.-Y. Zhu, T. Zhou, A. A. Efros, Image-to-image translation with conditional adversarial networks, 2017.

[28] S. Hicsonmez, N. Samet, E. Akbas, P. Duygulu, Ganilla: Generative adversarial networks for image to illustration translation, Image and Vision Computing (2020) 103886.

[29] X. Huang, M.-Y. Liu, S. Belongie, J. Kautz, Multimodal unsupervised image-to-image translation, in: IEEE European Conference on Computer Vision, 2018.

[30] J. Lee, E. Kim, Y. Lee, D. Kim, J. Chang, J. Choo, Reference-based sketch image colorization using augmented-self reference and dense semantic correspondence, in: IEEE Conference on Computer Vision and Pattern Recognition, 2020.

[31] C. Zou, Q. Yu, R. Du, H. Mo, Y.-Z. Song, T. Xiang, C. Gao, B. Chen, H. Zhang, Sketchyscene: Richly-annotated scene sketches, in: IEEE European Conference on Computer Vision, 2018.

[32] H. Kim, H. Y. Jhoo, E. Park, S. Yoo, Tag2pix: Line art colorization using text tag with secat and changing loss, in: IEEE International Conference on Computer Vision, 2019.

[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick, Microsoft COCO: Common objects in context, in: IEEE European Conference on Computer Vision, 2014.

[34] E. Kim, S. Lee, J. Park, S. Choi, C. Seo, J. Choo, Deep edge-aware interactive colorization against color-bleeding effects, in: IEEE International Conference on Computer Vision, 2021.

[35] H. Caesar, J. Uijlings, V. Ferrari, Coco-stuff: Thing and stuff classes in context, in: IEEE Conference on Computer Vision and Pattern Recognition, 2018.

[36] B. Zhou, H. Zhao, X. Puig, T. Xiao, S. Fidler, A. Barriuso, A. Torralba, Semantic understanding of scenes through the ade20k dataset, International Journal of Computer Vision 127 (3) (2019) 302–321.

[37] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, A. Lerer, Automatic differentiation in pytorch (2017).

[38] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization (2014). arXiv:1412.6980.

[39] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, S. Hochreiter, Gans trained by a two time-scale update rule converge to a local nash equilibrium, in: Advances in Neural Information Processing Systems, 2017.