Language-based Colorization of Scene Sketches

Supplementary Materials

Changqing Zou[1,2*] Haoran Mo[1*] Chengying Gao[1†]
Ruofei Du[3‡] Hongbo Fu[4]
Sun Yat-sen University[1] Huawei Noah's Ark Lab[2]
Google[3] City University of Hong Kong[4]

1 Technical Details

1.1 Loss function formulations

Foreground Colorization. Let x be an input object instance sketch image, y the corresponding ground truth image, and s the paired input natural language expression. The GAN objective function is expressed as:

$$L_{GAN}\left(D,G\right)=\mathbb{E}_{y\sim P_{image}}\left[\log D(y)\right]+\mathbb{E}_{x\sim P_{sketch},s\sim P_{text}}\left[\log(1-D(G(x,s)))\right], \tag{1}$$

and $L_{GAN}\left(G\right)$ uses the second term in this equation.

Let c be a class label output by the discriminator D. The auxiliary classification loss $L_{ac}(D)$ for D is defined as the log-likelihood between the predicted and the ground-truth labels:

$$L_{ac}(D)=\mathbb{E}\left[\log P(C=c|y)\right]. \tag{2}$$

The auxiliary classification loss $L_{ac}(G)$ for generator G is defined in the same form as $L_{ac}(G)=L_{ac}(D)$ with the discriminator fixed but the image to be classified as a synthesized one.

The supervision loss $L_{sup}(G)$ and the complete loss functions $L(D)$ and $L(G)$ for foreground colorization can be found in Equation 2, 3, and 4 of the main paper.

Background Colorization. Given the input image x with the partially or completely colorized foreground objects, the ground-truth color image y, and the language description s, the generator G produces the synthesized image with the colorized background $G(x,s)$. The cGAN objective function is expressed as:

$$L_{cGAN}\left(D,G\right)=\mathbb{E}_{x\sim P_{fg},y\sim P_{image}}\left[\log D(x,y)\right]+\mathbb{E}_{x\sim P_{fg},s\sim P_{text}}\left[\log(1-D(x,G(x,s)))\right], \tag{3}$$

and the objective of the generator $L_{cGAN}\left(G\right)$ is to minimize the second term.

Given the category size C, the segmentation mask prediction $R\in\mathbb{R}^{W\times H\times C}$, and the ground truth segmentation mask $\hat{R}$, the segmentation loss $L_{seg}(G)$ is expressed in a cross-entropy manner:

$$L_{seg}(G)=-\frac{1}{WH}\sum_{i=1}^{W}\sum_{j=1}^{H}\sum_{k=1}^{C}\left(\hat{R}_k^{ij}*\log(R_k^{ij})\right). \tag{4}$$

The supervision loss $L_{L1-sup}(G)$ and the complete loss functions $L(D)$ and $L(G)$ for background colorization can be found in Equation 5, 6, and 7 of the main paper.

---

*Both authors contributed equally to the paper.
†Corresponding author: mcsgcy@mail.sysu.edu.cn
‡This project was started before this author joined Google.

1.2 Implementation Details

Instance Matching Experiments. The maximum training iteration was $100K$ and the batch size was set to 1. The initial learning rate was set to 0.00025 and Adam [2] was used as the optimizer. We resized the scene sketch images and the corresponding ground-truth masks to 768×768. The iteration numbers of LSTM and mLSTM were both set to 15. The cell sizes of LSTM and mLSTM were respectively set to 1,000 and 500. The Deeplab-v2 model [1] was trained on the SketchyScene dataset [4].

Foreground Instance Colorization Experiments. We set the maximum training iteration to 100K and used a mini-batch size of 2. We employed Adam [2] as the optimizer and set the initial learning rate of generator to 0.0002 and that of discriminator to 0.0001. The iteration numbers of LSTM and mLSTM were both 15 and their cell sizes were both set as 512. We set $\lambda_1 = 1$ and $\lambda_2 = 100$ in Equations 3 and 4 in the main paper.

Background Colorization Experiments. We trained 100K iterations using a mini-batch size of 1. Adam optimizer was used and the initial learning rate for both generator and discriminator was set to 0.0002. The iteration numbers of LSTM and mLSTM were both 9 and their cell sizes were both 1024. We set both λ_1 and λ_2 at 100 in Equation 7 in main paper.

2 Data Collection Details

2.1 Data Collection for Instance Matching

To train the instance matching network, we require triplet samples of scene sketches, text descriptions, and instance mask(s) as shown in Figure 6 in the main paper. Since collecting such a kind of data through manual annotation requires enormous crowdsourcing efforts, we designed and implemented a fully automatic rule-based algorithm to generate the paired data, based on some insights we learnt from the SketchyScene data [4] and the human cognition as below:

- The 24 selected categories (as shown in Table 1 of the main paper) can be divided into several higher-level groups based on their characteristics, as shown in Table 1.

Table 1: Higher-level grouping of object categories.

Groups	Categories
Distant objects	sun, moon, cloud, star
Still objects	house, bus, truck, car, bench, tree, road, grass
Animated objects	bird, butterfly, cat, chicken, cow, dog, duck, horse, people, pig, rabbit, sheep

- Humans tend to describe the adjacent objects with the same category using a single expression, *e.g. "the two trees on the left are green"*.
- Humans tend to describe distant objects without other reference objects or spatial information, *e.g. "the clouds are light blue" / "all the stars in the sky are red"*.
- For still objects, humans tend to describe them without other reference objects but with optional spatial information, *e.g. "the left house is red with black roof" / "all the grass are dark green" / "the road is black"*.
- For animated objects, humans tend to describe them with still objects as reference along with optional spatial information, *e.g. "the person near the left car is in blue" / "the second chicken on the right is yellow" / "the dog has brown body"*.

Based on these insights, we designed a fully automatic rule-based algorithm, which is summarized in Algorithm 1. In this algorithm, we obtained the language expression describing the location of an instance, *e.g. "the tree in the middle" / "the bus"*, as well as its binary mask as shown in Figure 6 of the main paper. However, in practice, the instructions that users assign to the system specify not only the instance of their interest, but also their colorization goal, such as *"the tree in the middle is green"*. To construct such a fully automatic model which still works well on distinguishing specified target(s) based on an expression even with extra colorization information, we turned to augmenting the location-only expression with random colorization descriptions. For example, after obtaining *"the bus"*, we randomly selected a colorization description designed for *bus*, *e.g. "has orange body and blue windows"*, from the *FOREGROUND* dataset, thus producing *"the bus has orange body and blue windows"* finally. Note that data collection for the instance matching task was automatically completed without any manual annotation.

Algorithm 1: Instance Matching Data Generation

Input: bboxes $\mathbf{B} : [B_1, B_2, ...B_n]$, class_labels $\mathbf{L} : [L_1, L_2, ...L_n]$, masks $\mathbf{M} : [M_1, M_2, ...M_n]$

Output: a set of $\mathbf{O}$ {caption T: its corresponding masks $[M_p, M_q, ...]$}

```
for B, L ∈ B, L do
    raw_items = RegisterItem(B,L)

distant_items = SelectDistantItems(raw_items)
O_dist {T_dist : [M_p, M_q, ...]} = GetTextAndMasksByItemNumber(distant_items, M)

near_items = SelectNeartItems(raw_items)
still_items, animated_items = SplitItems(near_items)

Function GroupingAdjacentItems(items):
    for item ∈ items do
        recursively look for another item_t ∈ items
        if IsSameCategory(item_t, item) & NotGrouped(item_t) &
          IsAdjacent(item_t, item) then
            item_groups = MakeItemGroups(item_t, item)
            item_groups_map = {item_groups.category: item_groups}
    return item_groups_map;

still_groups = GroupingAdjacentItems(still_items)
animated_groups = GroupingAdjacentItems(animated_items)

Function SetPositionOfItemsWithinGroup(group):
    SortByHorizontalPos(group)
    pos_distribution = FindPosDistribution(group)
    for item ∈ group do
        item.SetPosition(pos_distribution)
    return;

Function FindReference(self_groups, ref_groups):
    SortByHorizontalPos(self_groups)
    for s_group ∈ self_groups do
        if IsEmpty(ref_groups) then
            ref = FindClosestRefWithinSelfGroup(self_groups)
        if IsNotEmpty(ref_groups) then
            ref = FindClosestRefWithinRefGroup(ref_groups)
        s_group.SetReference(ref)
        SetPositionOfItemsWithinGroup(s_group)
    return;

FindReference(still_groups, [ ])
FindReference(animated_groups, still_groups)

O_near {T_near : [M_p, M_q, ...]} = GetTextAndMasksByRefAndPos(still_groups +
  animated_groups, M)

O = O_dist + O_near
```

2.2 Data Collection for Foreground Instance Colorization

The foreground instance colorization task requires triples of cartoon image, edge map (sketch), language description, as shown in Figure 7 of the main paper. The detailed procedure of data collection for this task is described below:

1. We first crawled cartoon instance images, covering 24 object categories, from the Internet and then leveraged X-DoG [3] to extract an edge map as the corresponding sketch for each image. All the cartoon images and sketches were resized to 192×192. We split the data into the training and validation sets. As mentioned in Section 6 of the main paper, we also built a test set which consisted of instance sketches from the SketchyScene [4] dataset. The detailed numbers of examples for each category are shown in Table 2.

Table 2: Detailed information for foreground instance data.

Category	Train	Val.	Test	Category	Train	Val.	Test
bench	119	24	50	bird	182	37	100
bus	167	33	34	butterfly	172	34	50
car	172	34	150	cat	223	45	50
chicken	164	33	100	cloud	132	26	50
cow	178	36	50	dog	165	33	50
duck	168	34	50	grass	109	22	50
horse	151	30	50	house	208	41	200
moon	124	25	50	people	252	51	200
pig	135	27	50	rabbit	160	32	50
road	100	20	50	sheep	155	31	50
star	167	33	50	sun	152	30	50
tree	139	28	50	truck	128	26	100
				Total	3822	765	1734

2. Before collecting the color descriptions, we pre-defined 16 commonly used colors as shown in Table 3, and the semantic part hierarchies for all the 24 categories as in the dataset for instance matching as shown in the "Parts" column in Table 4. We pre-defined the semantic part hierarchies because of the observation that some categories can be entirely described in a single color, while others tend to have different colors for different object parts (*e.g.,* the windows and the body of a car might have different colors). For the latter ones, we need to assign part-level colors.

Table 3: Pre-defined colors for foreground objects.

	Colors
Foreground	red, orange, yellow, light green, dark green, cyan, light blue, dark blue, purple, pink, black, light gray, dark gray, light brown, dark brown, white

3. Based on the above preparation for color description collection, we designed an effective approach with the aid of both human manual annotation and automatic generation, which reduced significantly the human effort compared with fully manual annotation.

4. At the human manual annotation side, we designed an easy way for users to make color annotations. For example, to generate the descriptions for the colors of a car and its windows,

we firstly made two folders named with *"body"* and *"windows"*. Inside the two folders, we each made 16 empty folders named with the color phrases shown in Table 3. Then, workers only needed to drag-and-drop the collected car images to the 16 empty folders for each part (*"body"* or *"windows"*) according to the color of the specified part.

5. At the automatic generation side, we first pre-designed some description patterns for each of the 24 categories according to its semantic part hierarchy, as shown in Table 4. After the human manual annotation, the descriptions were automatically generated with the these sentence patterns.

Table 4: Description patterns for foreground categories.

Category	Parts	Description patterns
bench, butterfly, cat, cloud, cow, dog, duck, grass, horse, moon, pig, rabbit, road, sheep, star, sun, tree	Single	*"the ...(category) is ...(color)"*
bird	body, wing	*"the bird is ..."* *"the bird has ... body"* *"the bird is ... with ... wing"* *"the bird has ... body and/with ... wing"*
chicken	body, head, tail	*"the chicken is ..."* *"the chicken has ... head and/with ... body"* *"the chicken has ... body and/with ... tail"* *"the chicken has ... head, ... body and/with ... tail"*
bus	body, windows	*"the bus is ..."* *"the bus is ... with ... windows"* *"the bus has ... body and/with ... windows"*
car	body, windows	*"the car is ..."* *"the car is ... with ... windows"* *"the car has ... body and/with ... windows"*
truck	body, carriage	*"the truck is ..."* *"the truck is ... with ... carriage"*
house	body, roof	*"the house is ..."* *"the house is ... with ... roof"*
people	hair, shirt, pants/ skirt	*"the person is in ..."* *"the person has ... hair and is in ..."* *"the person is in ... shirt and/with ... pants/skirt"* *"the person has ... hair and is in ... shirt and/with ... pants/skirt"*

6. To imitate user inputs in practice, which might contain both location and colorization information, we randomly augmented the location information based on sentence structure patterns for each collected description. For example, in Figure 7 of the main paper, after obtaining *"the chicken is light brown"* by the above steps, we randomly selected a location phrase from the *MATCHING* dataset, *e.g.* *"in front of the house"*, and inserted it between *"the chicken"* and *"is light brown"*. This can be done since we have already known the sentence structures as summarized in Table 4. Thus, we obtained the complete description *"the chicken in front of the house is light brown"*. Note that this augmentation is optional, because users might not always assign instructions with location information. For example, given a scene sketch with only one car, users probably assign a simple instruction like *"the car is/has ..."* without describing its location.

With the above procedures, we employed 6 users to annotate, through the drag-and-drop way, the colors of the overall or part-level regions of the cartoon images, and then obtained the description sentences automatically .

2.3 Data Collection for Background Colorization

Figure 1: Illustration of the data collection procedure for background colorization.

The pipeline of the data collection for background colorization is shown in Figure 1 (the same as Figure 8 in the main paper), which produces four modality data: foreground image, background-colorized image, description, and segmentation label map. The detailed procedure is as follows:

1. Since the SketchyScene [4] dataset has provided the ground-truth bounding box (sketch template, Figure 1(a)) of each instance, we first searched our cartoon clip art dataset for the cartoon instances with the same category and similar size to each bounding box and then placed them into a 768×768 white canvas, which forms the foreground image, as shown in Figure 1(b).
2. We recruited users to produce the background-colorized images by manually painting the blank regions with solid colors with practical color filling tools such as the *Paint* tool under Windows. Specifically, we required users to paint with only two colors, "blue" (in RGB (153, 217, 234)) as *sky* and "green" (in RGB (181, 230, 29)) as *ground*, as shown in the fourth column of Figure 1.
3. Since we have known the distinct RGB values of the *sky* and the *ground*, we obtained the segmentation mask of three categories: *sky*, *ground* and *foreground* simply by checking the color value of each pixel, as shown in Figure 1(c).
4. With the segmentation mask, we first defined several color phrases with different RGB values (11 colors for *sky* and 5 colors for *ground*, as shown in Table 5), and then randomly assigned the colors to the *sky* and *ground* regions as a data augmentation process for each foreground image. Given the randomly selected colors, we produced the descriptions based on the pattern *"the sky is ... and the ground is ..."*, as shown in the three columns on the right of Figure 1. Note that the data augmentation and the description generation can both be done automatically, thus making it possible to generate a large-scale dataset.

Table 5: Color definition for background.

	Colors
Sky	red, orange, yellow, green, cyan, blue, purple, pink, black, gray, brown
Ground	yellow, green, black, gray, brown

With the designed procedures above, we first collected 3932, 300, and 727 sketch templates from the training, validation and test set of the SketchyScene dataset, and then produced foreground images for each template. Afterwards, we employed 24 users to produce a background-colorized image (all in *blue sky* and *green ground*) for each foreground image. Finally we automatically augmented each foreground image with 3 more background-colorized images, and totally obtained 15728, 1200, 2908 quadruple data for training, validation, and testing.

3 More Colorization Results

3.1 Un-targeted Colorization

Figure 2 shows more interactive results from the un-targeted colorization study of our system. These results cover instructions with a large degree of diversity, some of which are out of the coverage of the training data mentioned in the main paper, such as *"wild" sentence structure* (*e.g.*, "light green bus with blue windows" (A3), "red moon in the sky" (E4)), *language grammar* (*e.g.*, "all the clouds dark gray" (A4), "all the stars is red" (C4)), *unsupported words* (*e.g.*, the verb "make" in "make the sky blue and ground green" (A5) never appears in the training data).

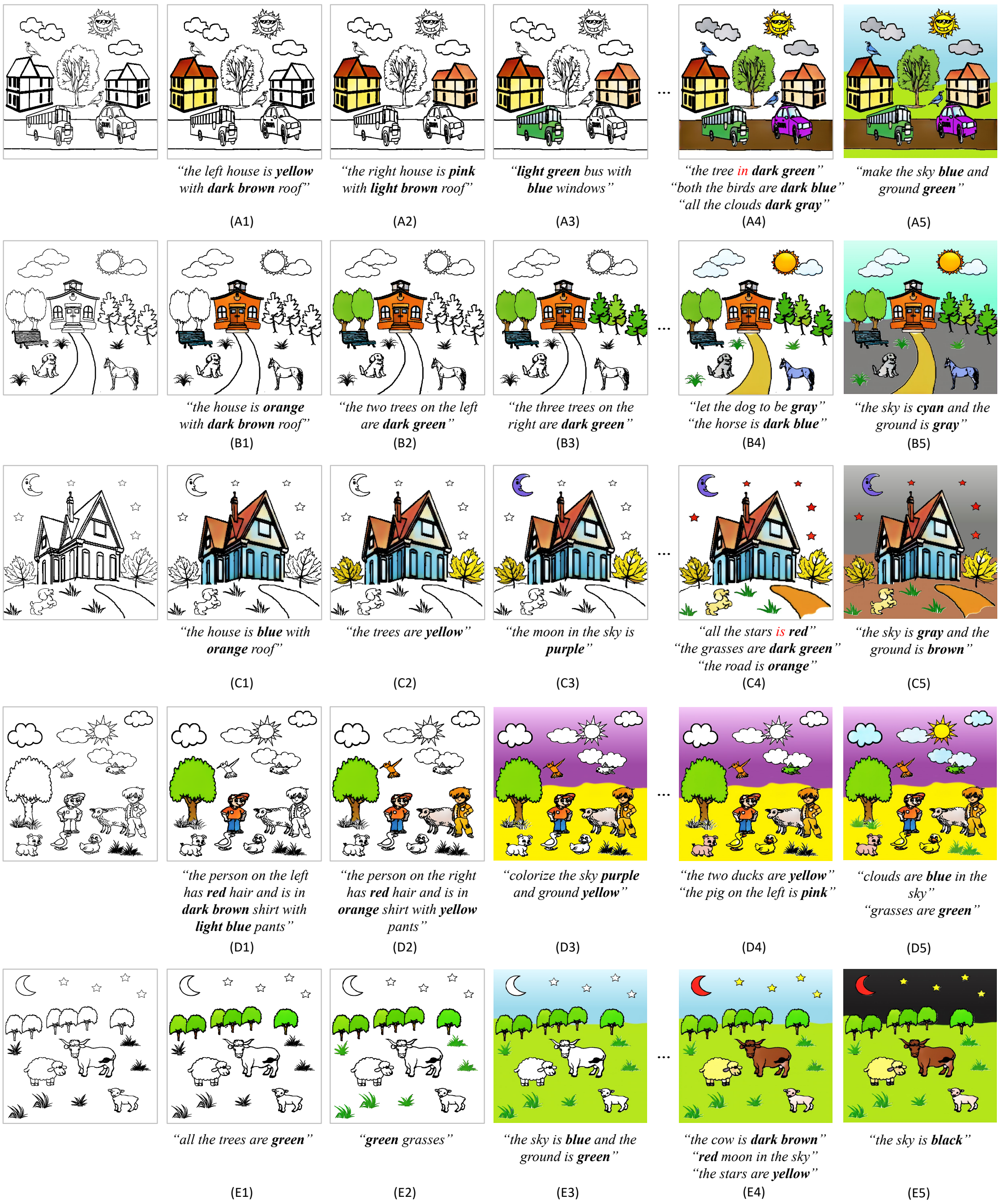
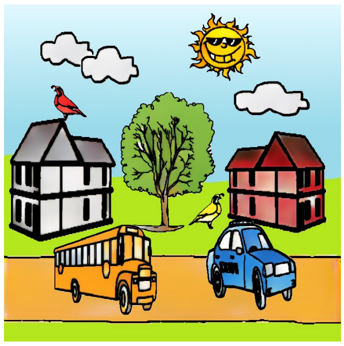


Figure 2: More interactive results from the un-targeted colorization study of our system.

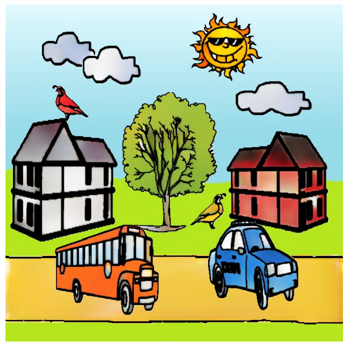
3.2 Targeted Colorization

Figures 3 to 11 show more results from the targeted colorization study of our system. We invited six users (A: 10-year-old boy in primary school. B: 21-year-old female graduate student. C: 23-year-old male graduate student. D: 22-year-old female graduate student. E: 14-year-old boy in high school. F: 30-year-old female working in a company.) to provide the input instructions for this study. In fact, different users might colorize the targets (foreground objects or background regions) in different orders. While in order to demonstrate the comparisons between instructions towards the same target, we arrange them in the same order. To allow better visualization, we highlight the different *expressions* towards the same target in **red** and different *color goals* in **blue**.

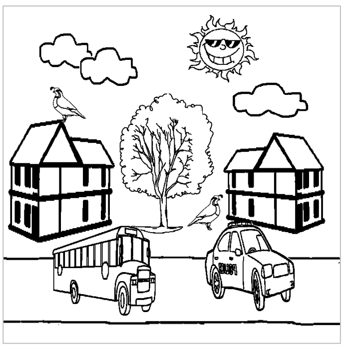
Target cartoon



Output from user C



Output from user F



Scene sketch

the tree is dark green

the clouds are ***light*** *gray*

the bus is orange

the car is dark blue

the bird ***on the bottom*** *is* ***light*** *yellow*

the left house is gray with black roof

the right house is red with black roof

the sun is orange

the bird ***on the left house*** *is red*

road is ***orange***

the sky is blue and the ground is ***light*** *green*

Input from user C

the tree is dark green ***with light brown trunk***

gray clouds ***are floating on the sky***

the orange bus ***has light blue windows***

the car is dark blue

the yellow bird ***is standing on the ground***

the left house is ***light*** *gray with a black roof*

the right house is red with a black roof

the sun ***is shining with orange rays***

the left bird is red

the road is ***yellow***

a blue sky and green ground

Input from user F

Figure 3: More results of the targeted colorization study.

References

[1] Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. Deeplab: Semantic Image Segmentation With Deep Convolutional Nets, Atrous Convolution, and Fully Connected Crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence 40*, 4 (2018), 834–848.

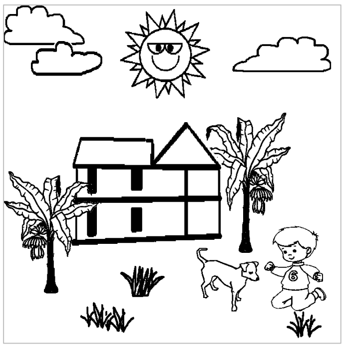
[2] Kingma, D. P., and Ba, J. Adam: A Method for Stochastic Optimization. *CoRR abs/1412.6980* (2014).

[3] Winnemöller, H. XDoG: Advanced Image Stylization With EXtended Difference-of-Gaussians. In *Eurographics Symposium on Non-Photorealistic Animation and Rendering* (2011), NPAR '11, ACM, pp. 147–156.

[4] Zou, C., Yu, Q., Du, R., Mo, H., Song, Y.-Z., Xiang, T., Gao, C., Chen, B., and Zhang, H. SketchyScene: Richly-Annotated Scene Sketches. In *ECCV* (New York, NY, USA, 2018), Springer International Publishing, pp. 438–454.

Figure 4: More results of the targeted colorization study.

Target cartoon | Output from user C | Output from user D



Scene sketch

*the sun **in the sky** is orange*

the clouds are light blue

*all the trees are dark green **with brown trunks***

*all the grasses are **dark** green*

*the person has a **red** hair and is in **yellow** shirt with blue pants*

the house is red with gray roof

*the dog on the right is **dark yellow***

the sky is pink and the ground is black

Input from user C

the sun is orange

the cloud is light blue

all the trees are green

all the grasses are green

*the person has **dark brown** hair and is in **orange** shirt and **dark** blue pants*

*the house is red with **dark** gray roof*

*the dog on the right is **light brown***

the sky is pink and the ground is black

Input from user D

Target cartoon | Output from user E | Output from user F

Scene sketch

the house is yellow with red roof

***one duck on the left** is purple*

***the other duck on the right** is white*

the road is yellow

the clouds are blue

trees are green

*sun is **yellow***

*grass **is** dark green*

sky is blue and ground is green

Input from user E

*the house with red roofs **has yellow doors***

***the left duck** is purple*

***the right duck** is white*

the road is yellow

*the clouds are **dark** blue*

*all the trees are **light** green*

*the sun is **orange***

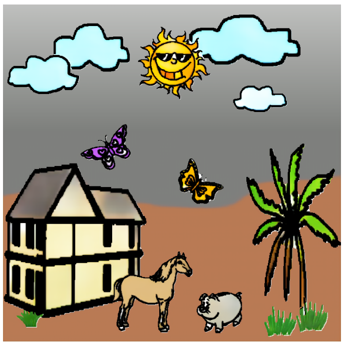
***all the** grasses **are** dark green*

***the** sky is blue and **the** ground is green*

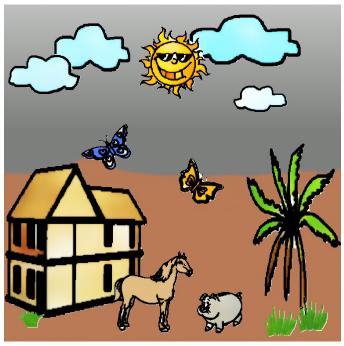
Input from user F

Figure 5: More results of the targeted colorization study.

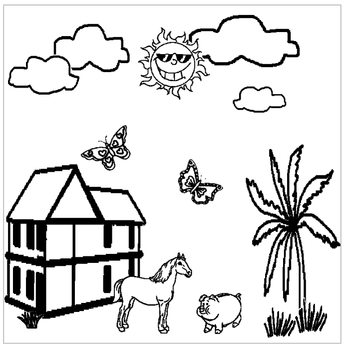
Target cartoon



Output from user D



Output from user E



Scene sketch

the tree is green

all grass are ***dark*** *green*

the sun is orange

the cloud is blue

the left butterfly is ***purple***

another *butterfly on the right is orange*

the horse on the left is brown

the pig is gray

the house is ***yellow with gray roof***

the sky is gray and the ground is brown

Input from user D

the tree is green

the grass is green

the sun is orange

all the clouds are blue

the butterfly on the left is ***dark blue***

the butterfly ***on right side*** *is orange*

the horse on the left is brown

the pig is gray

the ***front*** *house is* ***brown and yellow***

sky is gray and ***floor*** *is brown*

Input from user E

Target cartoon

Output from user C

Output from user F

Scene sketch

the road is ***dark*** *yellow*

the stars are yellow

the moon is black

the trees are ***dark*** *green*

the grasses are dark green

the dog is light brown

the house is ***yellow*** *with red roofs*

the sky is blue ***and*** *the ground is* ***light*** *green*

Input from user C

the road is yellow

the stars are yellow

the moon is black

the trees are ***light*** *green*

the grasses are dark green

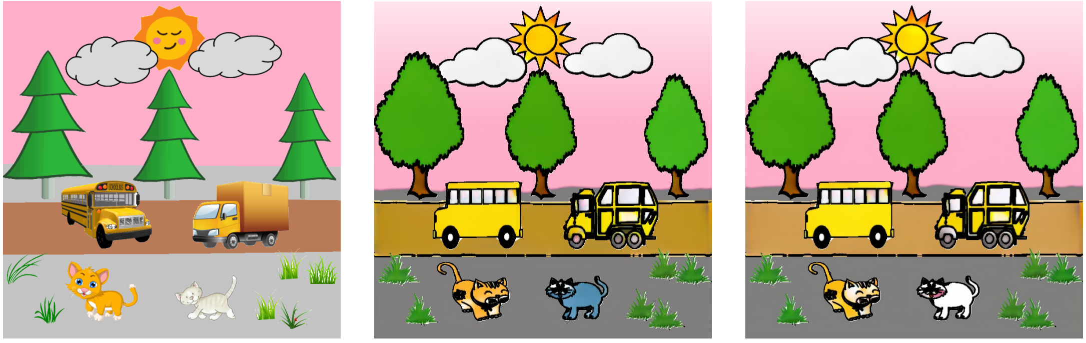
the dog is light brown

the ***orange*** *house* ***has red roofs and light blue windows***

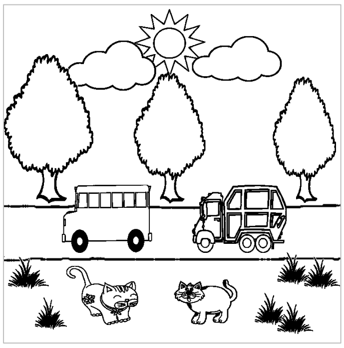
the sky is ***light*** *blue. the ground is green.*

Input from user F

Figure 6: More results of the targeted colorization study.



Target cartoon | Output from user C | Output from user F



Scene sketch

Input from user C:

*the sun is **orange***

the bus and the truck are yellow

all the trees are dark green

all the grasses are dark green

the clouds are gray

the road is brown

the left cat is orange

*the cat on the right is **cyan***

the sky is pink and the ground gray

Input from user F:

*the sun is **yellow with orange rays***

yellow bus

truck on the right is yellow

the trees are dark green

*the grasses are **also** dark green*

the clouds are gray

*the **light** brown road*

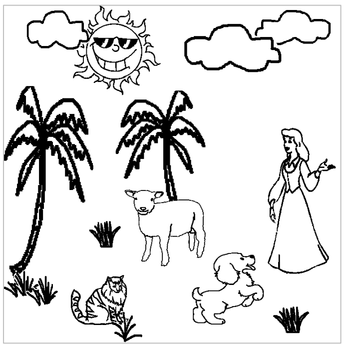
*the left cat is orange **with blue eyes***

*the cat on the right is **white***

the sky is pink and the ground is gray

Input from user C | Input from user F

Target cartoon | Output from user A | Output from user B



Scene sketch

Input from user A:

green/brown tree

***yellow** sun*

gray cloud

brown sheep

green grass

gray cat

***dark yellow** dog on right*

black person in a sut

blue sky. green ground.

Input from user B:

there are two trees, where the leafs are green, the trunks are brown

*the sun is **organge***

gray clouds on the sky

*the sheep is **dark** brown, **where the mouth is pink***

the grass is green

*the cat is **light** gray*

*the dog on the right is **light brown**, **and paint the ring on its nick is red***

*the person **has gray hair** and in black suit. **her shoes are black***

***paint** the sky blue and the ground with **light** green*

Input from user A | Input from user B

Figure 7: More results of the targeted colorization study.

Figure 8: More results of the targeted colorization study.

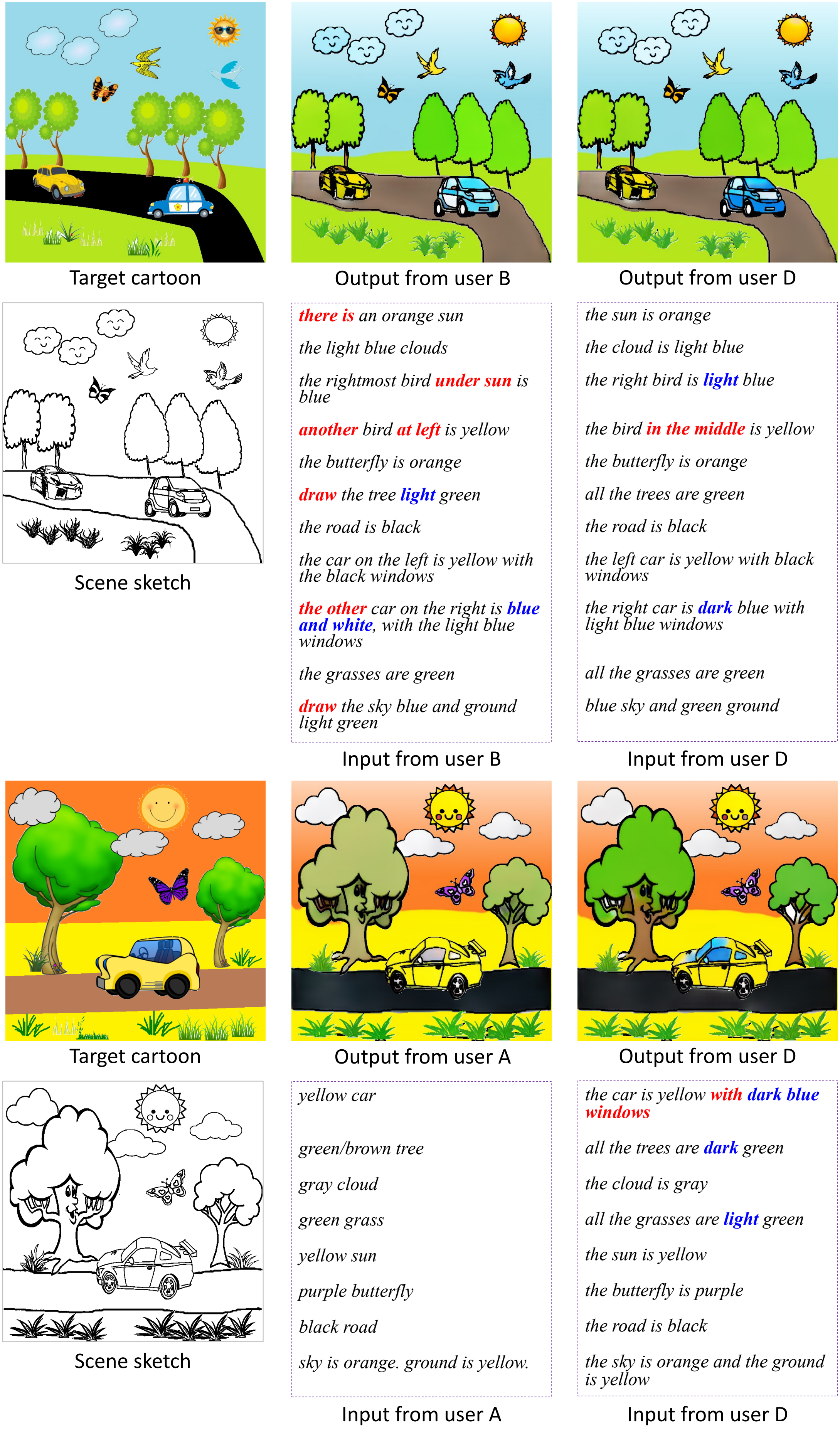


Figure 9: More results of the targeted colorization study.

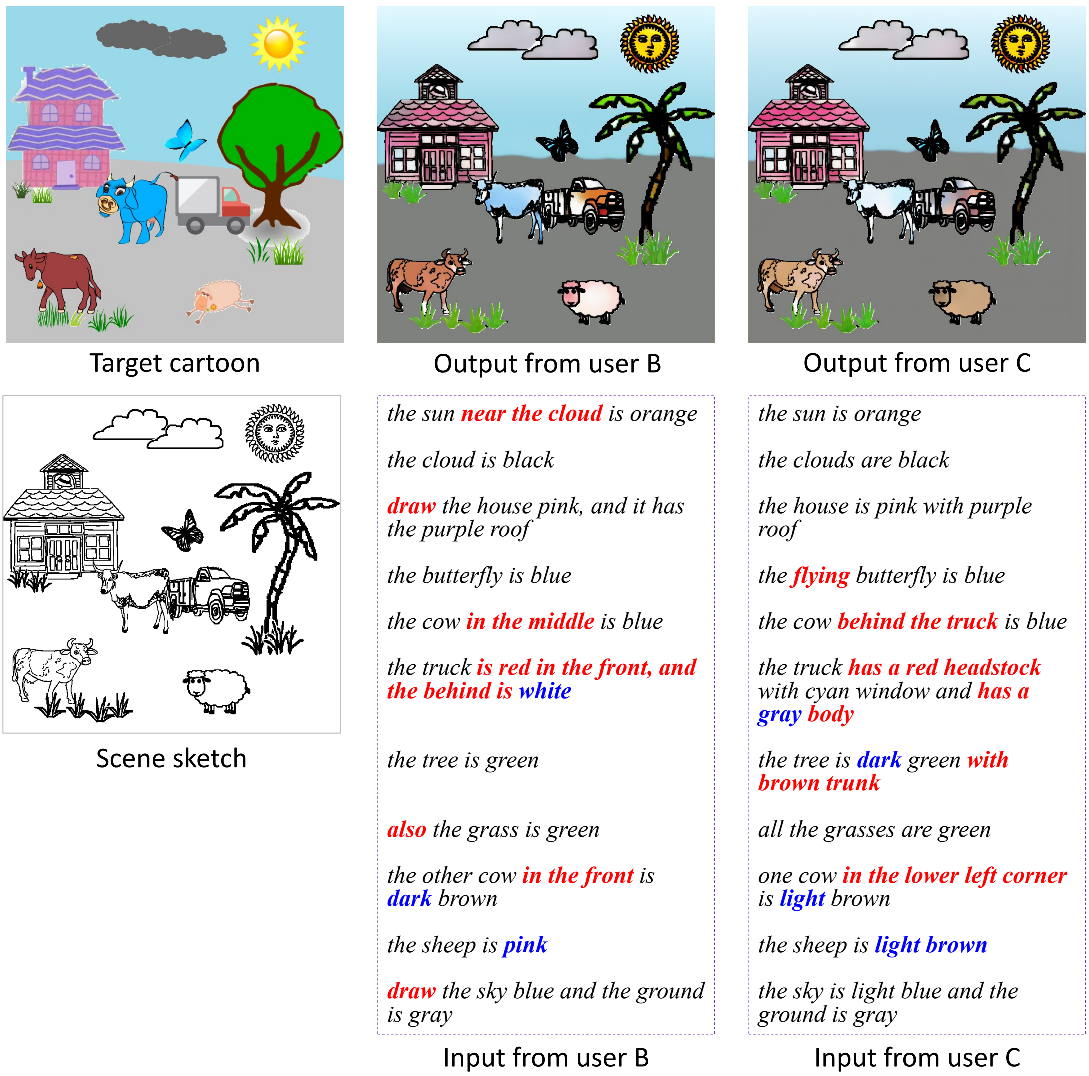


Figure 10: More results of the targeted colorization study.

Target cartoon

Output from user A

Output from user F

Scene sketch

Input from user A	Input from user F
blue cloud	*some* ***light*** *blue clouds* ***are floating in the sky***
green/brown tree	*four green trees* ***stand on the ground***
green grass	*the grasses are* ***dark*** *green*
yellow *sun*	*the* ***orange*** *sun* ***has a glass***
brown dogs	*the dog* ***near the house*** *is* ***light*** *brown*
	the dog ***on the leftmost*** *is* ***light*** *brown* ***with some dark grown spots***
	the dog ***on the bottom*** *is totally* ***dark*** *brown*
blue bird on leftmost	*the* ***light*** *blue bird on the left* ***has a pair of dark blue wings***
green bird on right	*the* ***light*** *green bird* ***flying on the right has a pair of dark green wings***
yellow and red house	*the house is yellow, and the roof is red*
purple sky. ground gray.	*the sky is purple and the ground is gray*

Figure 11: More results of the targeted colorization study.