Consistent Comic Colorization with Pixel-wise Background Classification

Sungmin Kang *
KAIST

Jaegul Choo
Korea University

Jaehyuk Chang
NAVER WEBTOON Corp.

Abstract

Comic colorization is a time-consuming task, acting as a bottleneck in comic drawing. We propose an automatic coloring model based on the observation that the background colors of comics are often consistent but random. From this observation, we introduce a novel background detector that learns to segment backgrounds out even without direct human annotation. This allows the generation of background-consistent colorizations, a feat that previous work fails to achieve.

1 Introduction

Colorization is a labor-intensive task in comic generation [5]. In this work, we propose a model designed specifically to colorize the images with a given outline in a realistic and consistent manner. The main novelty of our model lies in a novel automatic background detection network that can significantly improve the challenging task of correctly recognizing the background regions and painting them with a consistent color.

Previous Work. Recently, deep neural networks have seen great success in automatic colorization tasks of grayscale images [8, 9, 1]. Previous work also proposed methods for the colorization of comics [5, 4, 2, 11].

2 Proposed Model

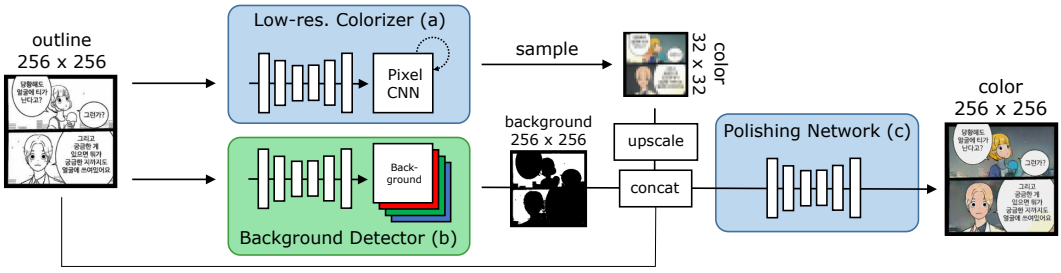


Figure 1: Overall architecture of the proposed comic colorization model. Our novelty lies in the background detector, highlighted in green. Note that this is a diagram of the inference precedure; each component of the model is trained separately.

As shown in Fig. 1 (a), the low-resolution colorizer follows the idea of the conditioning and coloring network of [9]. Our model used a distinct training configuration compared to [9]; details can be found in the supplementary material.

*Work done as an intern at NAVER WEBTOON Corp. Email: stuatlittle@kaist.ac.kr

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

A unique module in our proposed model is the background detector (Fig. 1 (b)) that detects the background region by a special training process. Specifically, the model generates RGB color channels as well as two segmentation channels. The model uses its own colorization for pixels classified as foreground. For those pixels classified as background, our model computes the averaged color of those pixels from the ground truth colored image and assigns this single color value to all the background pixels. Combining these two methods of coloring yields a colorized image, and we train the background detector to minimize the $L1$ distance between this image and the ground truth image. In this process, the model learns to parse the background region. This is because within the same comic, the colors of the characters and the speech balloons recur over multiple images, thus it is relatively easy for our model to predict them; on the other hand, it is more difficult to determine the correct color of the background region since its color is generally arbitrary depending on the images. Thus the network segments out the background of the image to use the average ground truth color instead of attempting to color the background itself.

Finally, the polishing network (Fig. 1 (c)) aggregates the segmentation information and low-resolution information to generate a high-resolution image. Implementation details may be found in the supplementary material.

3 Experimental Results

Dataset. The images used for training and validating the model were taken from the webcomic *Yumi's Cells*.[2] Details about the data are provided in the supplementary material.

Figure 2: From left, outline (a), background detection (b), low-resolution colorization (c), colorization without background information (d), colorization with background information (e), and ground truth colorization (f). Best seen in color. © Donggeon Lee.

Results. Figure 2 shows the intermediate outputs and the final colorization result of validation images (i.e., which was not seen by the model during training phase). The background of the image is segmented out from the foreground objects (Fig. 2 (b)). The low-resolution colorization is generated (Fig. 2 (c)), but the background color is inconsistent at this point. Passed through the polishing network without background information, the inconsistency causes the images to look artificial (Fig. 2 (d)). With background information, the inconsistency is handled, and the resulting images look natural (Fig. 2 (e)). Additional coloring results are provided in the supplementary material.

[2] `http://comic.naver.com/webtoon/list.nhn?titleId=651673`

References

[1] Alexander Kolesnikov Amelie Royer and Christoph H Lampert. Probabilistic image colorization, 11 May 2017.

[2] Keisuke Ogaki Yuri Odagiri Chie Furusawa, Kazuyuki Hiroshiba. Comicolorization: Semi-automatic manga colorization. arXiv:1706.06759, 2017.

[3] Ben Poole Eric Jang, Shixiang Gu. Categorical reparameterization with gumbel-softmax. In International Conference on Learning Representations, 2017.

[4] Kevin Frans. Outline colorization through tandem adversarial networks. arXiv:1704.08834, 2017.

[5] Paulina Hensman and Kiyoharu Aizawa. cgan-based manga colorization using a single training image. arXiv:1706.06918, 2017.

[6] Itseez. Open source computer vision library. `https://github.com/itseez/opencv`, 2015.

[7] Tinghui Zhou Alexei A. Efros Phillip Isola, Jun-Yan Zhu. Image-to-image translation with conditional adversarial networks. In IEEE Conference on Computer Vision and Pattern Recognition, 2017.

[8] Alexei A. Efros Richard Zhang, Phillip Isola. Colorful image colorization. In European Conference on Computer Vision, 2016.

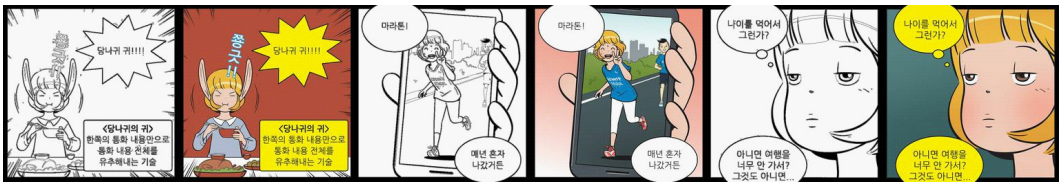
[9] David Bieber Mohammad Norouzi Jonathon Shlens Kevin Murphy Sergio Guadarrama, Ryan Dahl. Pixcolor: Pixel recursive colorization. arXiv:1705.07208, 2017.

[10] Xi Chen Diederik P. Kingma Tim Salimans, Andrej Karpathy. Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. arXiv:1701.05517, 2017.

[11] Taizan Yonetsuji. Paintschainer. `https://github.com/pfnet/PaintsChainer`, 2017.

Supplementary Material

Dataset. The images used for training and validating the model were taken from the webcomic *Yumi's Cells*,[3] from the opening episode to episode number 238. This consisted of 7394 images, which were randomly partitioned to a training set of 7014 images and a validation set of 380 images. The images were resized to a 256×256 pixel size. Outline images were generated from the color images by an algorithm that combines the canny edge detection results provided by OpenCV [6] and the black-colored part of a given image. All images shown in this paper are from the validation set. As seen in Figure 3, the background color is subject to the artist's whim and is thus difficult to determine from only the outline image. All *Yumi's Cells* comic images are under copyright of the artist Donggeon Lee, and were used in this paper with the artist's permission.



(a) (b) (a) (b) (a) (b)

Figure 3: The randomness of the background color in comics. Outline (a) and ground truth colorization (b) are presented together. © Donggeon Lee.

Low-resolution Colorizer. While we closely followed the architecture of [9], there are a few key differences. One is that while [9] pretrained their conditioning network using a separate segmentation dataset, we could not as there was no adequate dataset to pretrain our model on. Despite this disadvantage, our low-resolution colorizer still learns to colorize characters successfully. The other is that while [9] found the use of logistic mixture models [10] to be unnecessary, we find that for our dataset it is necessary to use them to improve the generalization capability of our model. This may be due to the small size of the dataset, and thus suggests that PixelCNN models perform better with logistic mixture models in the face of smaller datasets.

Background Detector. The background detector's architecture follows the generator architecture specified in [7]. There are two methods to generate the segmentation - either to use the sigmoid activation function at each pixel position to indicate the background, or to use the method of Gumbel-Softmax [3]. While there is no clear difference in performance, the segmentations generated by Gumbel-Softmax tend to be clearer. In any case, the foreground and background segment maps are 256×256 images with values between 0 and 1. We multiply the RGB channels with the foreground channel, and add this with the average pixel color multiplied by the background segment map to obtain a colorized image. The $L1$ distance between this image and the ground truth image is used as our loss function to train the background detector.

Polishing Network. The polishing network's architecture follows the generator architecture specified in [7]. Ten input channels are provided to the polishing network. Specifically, they are the masked low-resolution image (3 channels), the mask (1 channel), the outline image (1 channel), the foreground and background segmentations (2 channels), and background average color calculated from the low-resolution image (3 channels). During training time, we used the ground-truth low-resolution image, and used a trained background detector to generate background segments as we trained. We randomly masked the pixels of the low-resolution image during training time as having the model rely heavily on the low-resolution image was not desirable, following the rationale of [4] in a different manner. During the sample phase, we mask the low-resolution colorization that is perceived to be background; we instead provide the average color of the background region in the low-resolution colorization. The polishing network is necessary because the task of using the low-resolution image and the background segmentation to generate a high-quality colorization that matches the outline is not trivial. In addition, the low-resolution image fails to capture fine details of an image, and as such the polishing network is necessary for filling in narrow areas.

[3] Korean Version: `http://comic.naver.com/webtoon/list.nhn?titleId=651673`, English version: `http://www.webtoons.com/en/romance/yumi-cell/list?title_no=478`

Additional Colorization Results

(a) (b) (c) (d) (e) (f)

Figure 4: Comparison with baseline models. From left, outline (a), pix2pix [7] (b), Tandem Adversarial Networks [4] (c), PixColor [9] (d), our model (e), and ground truth image (f).
© Donggeon Lee.

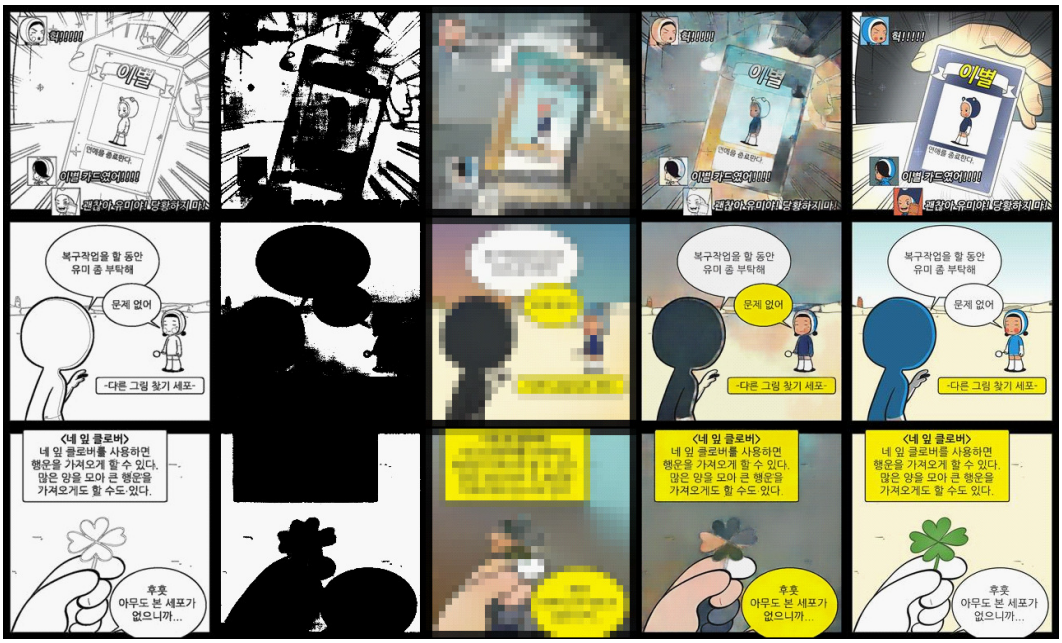


Figure 5: Failure cases. From left, outline (a), background detection (b), low-resolution colorization (c), final colorization (d), and ground truth image (e). On first row, background detection is noisy and inaccurate; on second row, background detector misclassifies, causing blush on ground; on third row, low-resolution colorizer colorizes hand and clover in inconsistent manner. © Donggeon Lee.