Generative Colorization of Structured Mobile Web Pages — Supplemental Material —

Kotaro Kikuchi[1] Naoto Inoue[1] Mayu Otani[1] Edgar Simo-Serra[2] Kota Yamaguchi[1]

[1]CyberAgent [2]Waseda University

1. Content Information Details

We represent a web page as an ordered tree, where each vertex is an element on the page and has content and color style information, as described in Section 3.1 of the main paper. A list of content information used in our experiments is shown in Table 1.

Table 1: List of content information and corresponding embedding methods used in our experiments. *Lookup* refers to a lookup translation to learnable embeddings, and *dense* refers to a linear mapping.

Name	Description	Domain	Embedding
Order	Element order within their siblings	$\mathbf{N}$	Lookup
Tag	HTML tag	{ "html", "body", "header", "button", $\cdots$ }	Lookup
Text	For elements with text, we extract features: number of lines and words, indicator variables for literal features (uppercase, capitals, numbers, and etc.). Zero vector otherwise.	$\left(\mathbf{N}^2 + \{0,1\}^9\right)$ or $\{0\}^{11}$	Dense
Image	For "img" tag elements, we extract features of the image referenced in "src" attribute: width, height, channel size, aspect ratio, mean and standard deviation in RGBA color, and whether it is SVG or not. Zero vector otherwise.	$\left(\mathbf{N}^3 + \mathbf{R}^9 + \{0,1\}\right)$ or $\{0\}^{13}$	Dense
Background image	Same as above, except the image comes from the computed value of "background-image" property.	$\left(\mathbf{N}^3 + \mathbf{R}^9 + \{0,1\}\right)$ or $\{0\}^{13}$	Dense

2. Details of Hierarchical Message Passing

As explained in Section 4.2 and Eq. (1)-(4) of the main paper, we encode content information of the elements with the bottom-up and top-down message passing [1]. Omitting the element index n for the variables for sets, he bottom-up message passing, $\mathrm{MP}_{\mathrm{up}} : \left(\{\bar{\mathbf{h}}_C\}, \mathbf{h}_{\mathrm{leaf}}; T\right) \mapsto \{\mathbf{h}_{\mathrm{up}}\}$ used in Eq. (2) of the main paper, is defined as

$$\mathbf{h}_{\mathrm{up}}^{(n)} = \begin{cases} \mathrm{MLP}_{\mathrm{up}}\left(\bar{\mathbf{h}}_C^{(n)} \oplus \mathbf{h}_{\mathrm{leaf}}\right) & \text{if } \mathrm{Child}\,(n;T) = \emptyset, \\ \mathrm{MaxPool}\left(\left\{\mathrm{MLP}_{\mathrm{up}}\left(\bar{\mathbf{h}}_C^{(n)} \oplus \mathbf{h}_{\mathrm{up}}^{(c)}\right)\right\}_{c \in \mathrm{Child}(n;T)}\right) & \text{otherwise,} \end{cases} \tag{1}$$

and the top-down message passing, $\mathrm{MP}_{\mathrm{down}} : (\{\mathbf{h}_{\mathrm{up}}\}, \mathbf{h}_{\mathrm{root}}; T) \mapsto \{\mathbf{h}_{\mathrm{down}}\}$ used in Eq. (3) of the main paper, is defined as

$$\mathbf{h}_{\mathrm{down}}^{(n)} = \begin{cases} \mathrm{MLP}_{\mathrm{down}}\left(\mathbf{h}_{\mathrm{up}}^{(n)} \oplus \mathbf{h}_{\mathrm{root}}\right) & \text{if } n \text{ is the root element of the tree } T, \\ \mathrm{MLP}_{\mathrm{down}}\left(\mathbf{h}_{\mathrm{up}}^{(n)} \oplus \mathbf{h}_{\mathrm{down}}^{\mathrm{Parent}(n;T)}\right) & \text{otherwise.} \end{cases} \tag{2}$$

In the equations above, we represent a multilayer perceptron as $\mathrm{MLP}(\cdot)$, the concatenation operator as $\oplus$, and the operations of extracting the children of a given element from the tree as $\mathrm{Child}(\cdot)$ and extracting the parent as $\mathrm{Parent}(\cdot)$, respectively.

3. Details of Core Generation Models

3.1. Autoregressive Model

We implement the model with both the Transformer encoder and decoder [4]. The Transformer encoder takes content embeddings as input and outputs hidden vectors. The Transformer decoder takes as input the embeddings of the estimated styles of the previous elements, and while attending to the hidden vectors, it estimates the color style for the next element. For the first style estimation in the decoder, a special learnable embedding is used as input instead of a style embedding. The model is trained by the teacher forcing [5], and at test time, it generates a color style for an element in one inference and repeats it for all the elements.

3.2. Non-autoregressive Model

We implement the model only with the Transformer encoder, which takes content embeddings as input and estimates the color styles of all the elements simultaneously.

4. Color Upsampler Details

As explained in Section 4.4 of the main paper, our color upsampler estimates the proportions in the quantization bins instead of the full resolution colors. Let $\boldsymbol{\alpha}$ be a vector representing the ground-truth proportions in the bins for all colors, the learning objective of the modified color upsampler model $\tilde{h} : (\mathcal{X}, \mathcal{C}, T) \mapsto \hat{\boldsymbol{\alpha}}$ is defined as:

$$\min_{\psi} E\left[\left(\tilde{h}\left(\mathcal{X}, \mathcal{C}, T; \psi\right) - \boldsymbol{\alpha}\right)^2\right], \tag{3}$$

where ψ is the model parameters. We implement this model with the Transformer encoder, which takes content and style embeddings as input.

5. Implementation Details

We set the same hyperparameters for all the Transformer networks: the feature dimension of input and output is 256, the number of attention heads is 8, the number of layers is 4, and the dimension of the inner feedforward network is 512. The layer normalization is performed on each layer before other operations [6]. We train our models with a batch size of 32 for 100,000 iterations, using the AdamW optimizer [3] with a learning rate of 1e-4.

6. Additional Visual Results

We present additional visual results in Figs. 1 to 5. Overall, the same trends as in the main paper can be observed: NAR and CVAE produce better color styles, and CVAE is capable of producing multiple variations. However, the results are not yet perfect and have several limitations. One of the typical failure cases discussed in Section 6 of the main paper is that elements with certain styles are not colorized. The right side example in Fig. 5 shows this failure case, as the icons and text of "Join as a Member..." are defined as rounded elements but no visible background colors are generated.

References

[1] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for Quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, ICML'17, 2017.

[2] Manoj Kumar, Dirk Weissenborn, and Nal Kalchbrenner. Colorization transformer. In *International Conference on Learning Representations*, ICLR'21, 2021.

[3] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, ICLR'19, 2019.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NeurIPS'17, 2017.

[5] Ronald J. Williams and David Zipser. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural Computation*, 1(2), 1989. doi: 10.1162/neco.1989.1.2.270.

[6] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20, 2020.

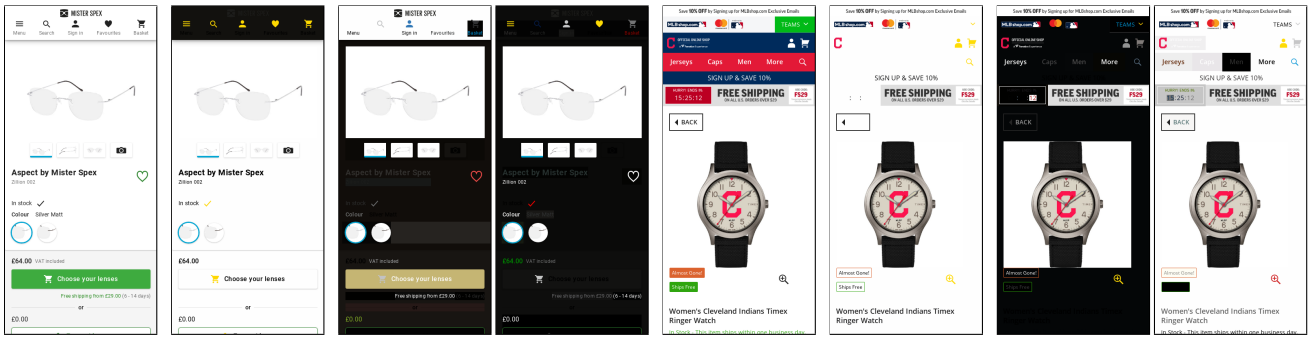
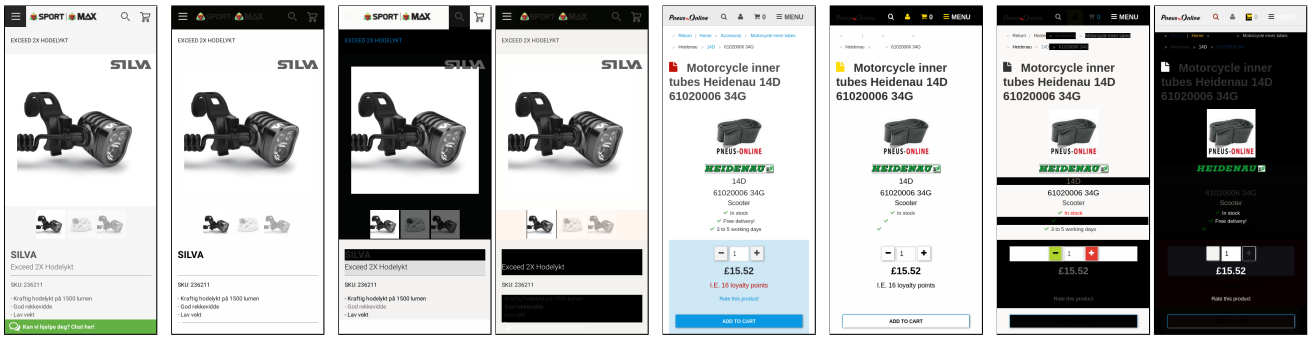
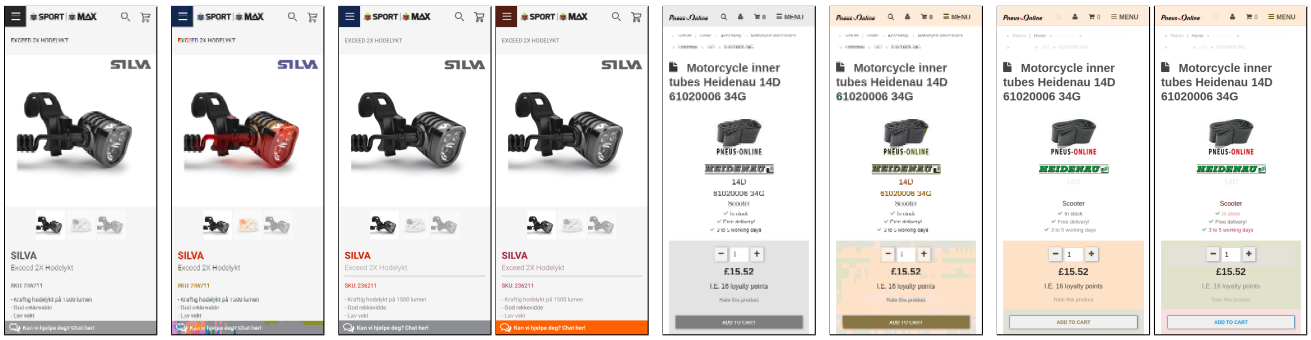


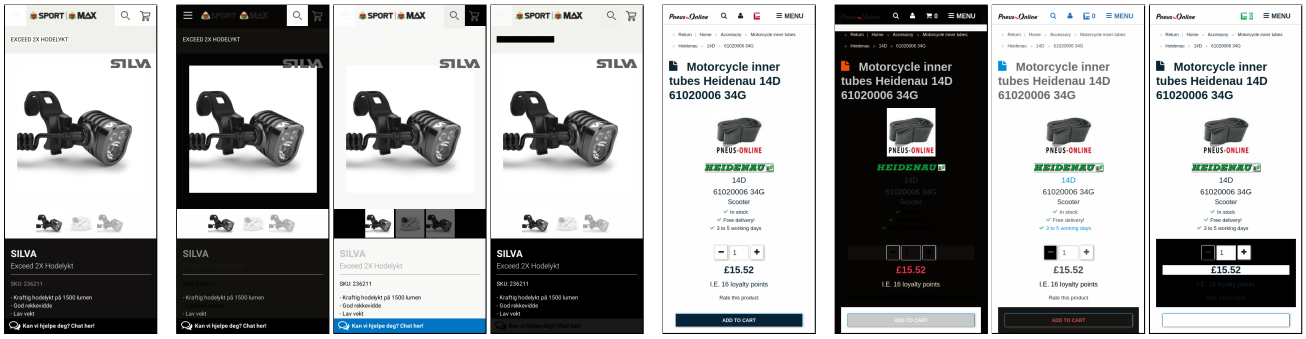
Figure 1: Additional qualitative results (1).



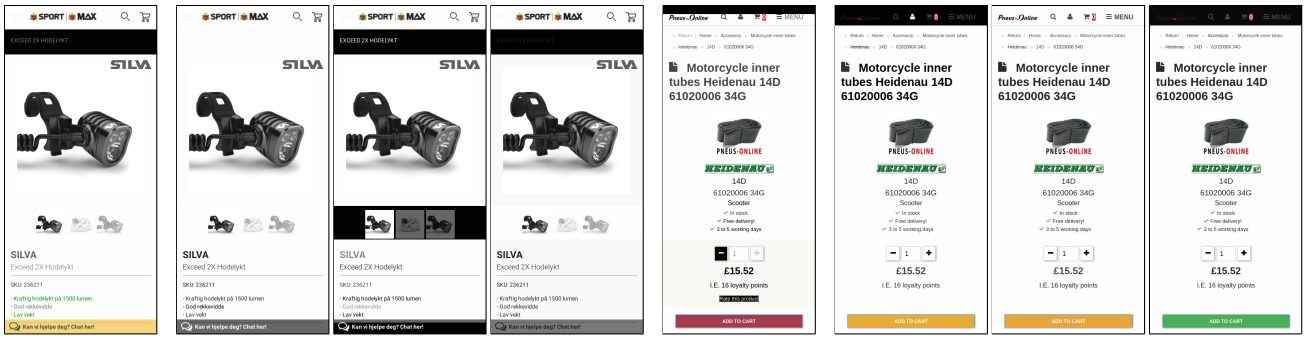
Ground-truth | Stats (mode) | Stats (sampling) | Ground-truth | Stats (mode) | Stats (sampling)



Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2] | Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2]

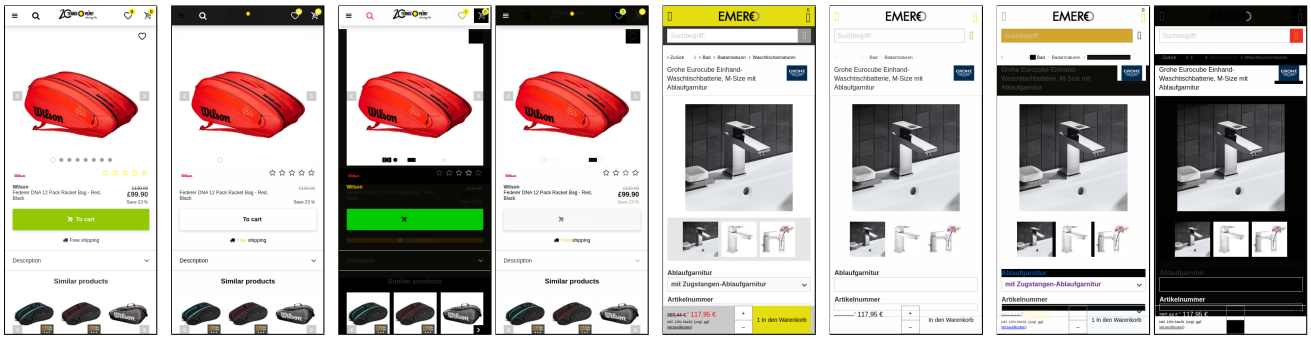


AR (greedy) | AR (top-p, p=0.9) | AR (greedy) | AR (top-p, p=0.9)

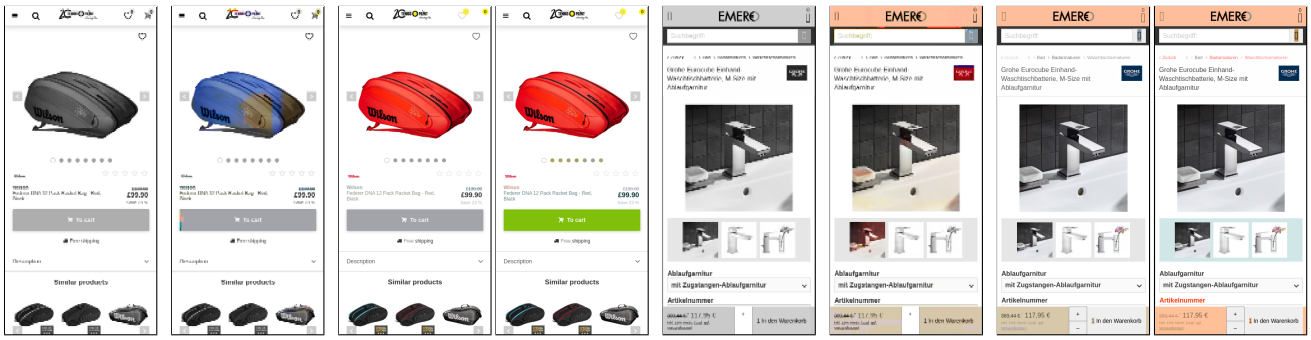


NAR | CVAE | NAR | CVAE

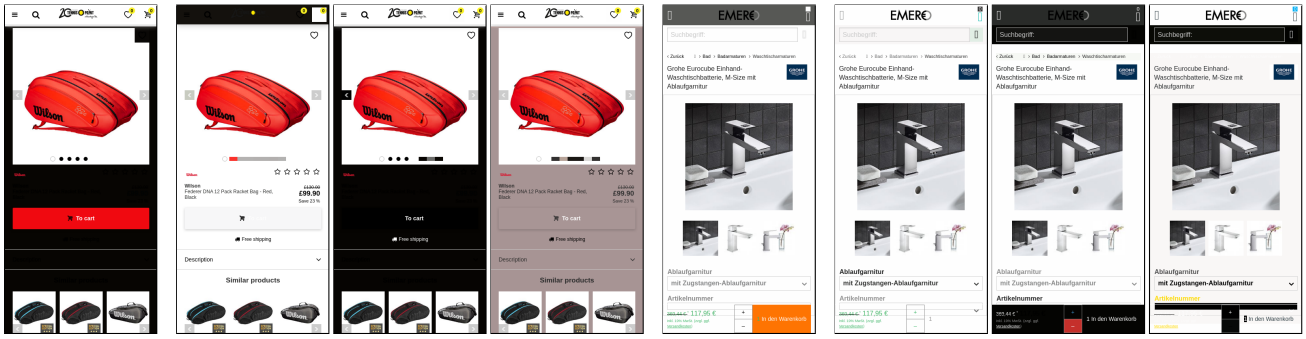
Figure 2: Additional qualitative results (2).



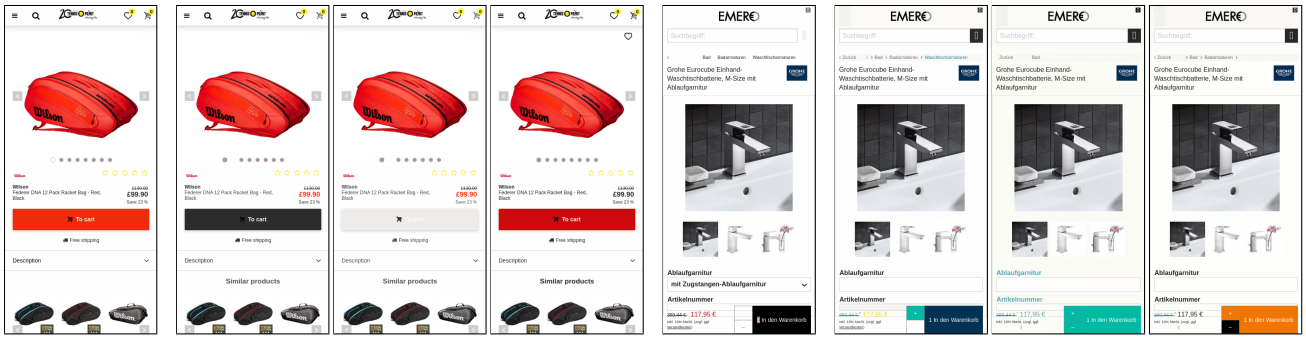
Ground-truth | Stats (mode) | Stats (sampling) | Ground-truth | Stats (mode) | Stats (sampling)



Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2] | Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2]

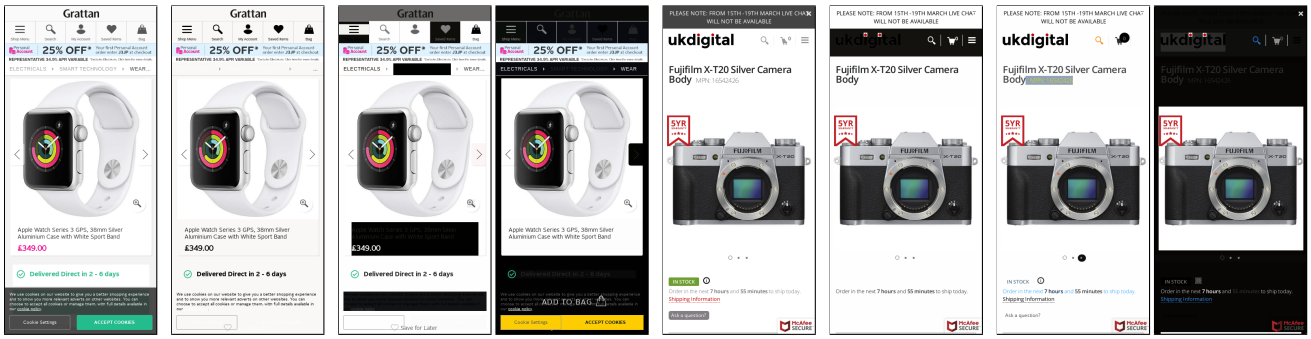


AR (greedy) | AR (top-p, p=0.9) | AR (greedy) | AR (top-p, p=0.9)

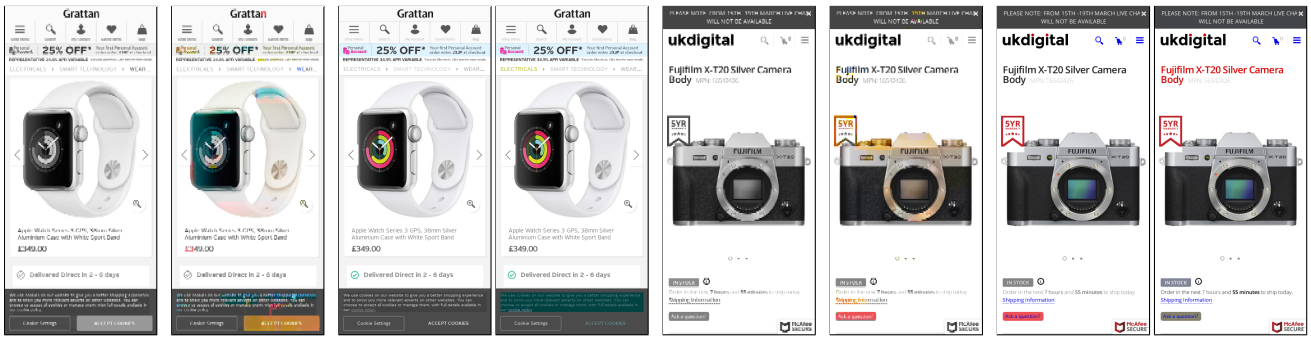


NAR | CVAE | NAR | CVAE

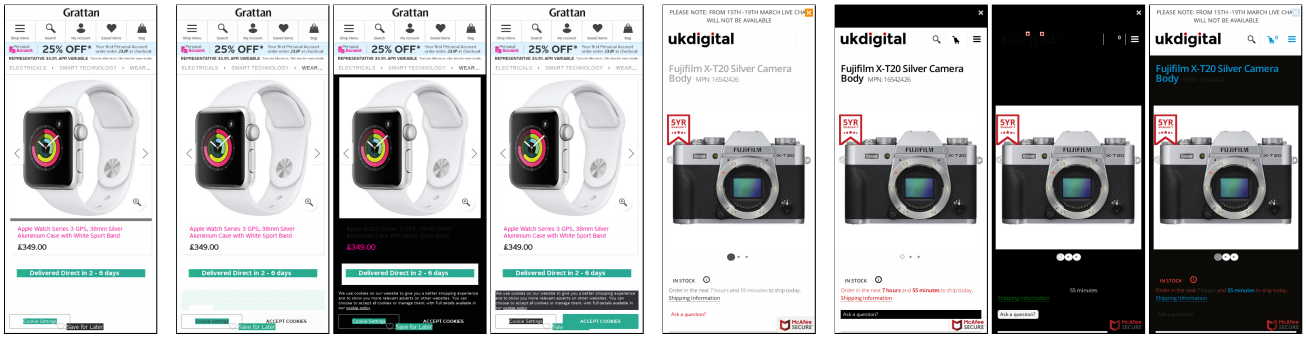
Figure 3: Additional qualitative results (3).



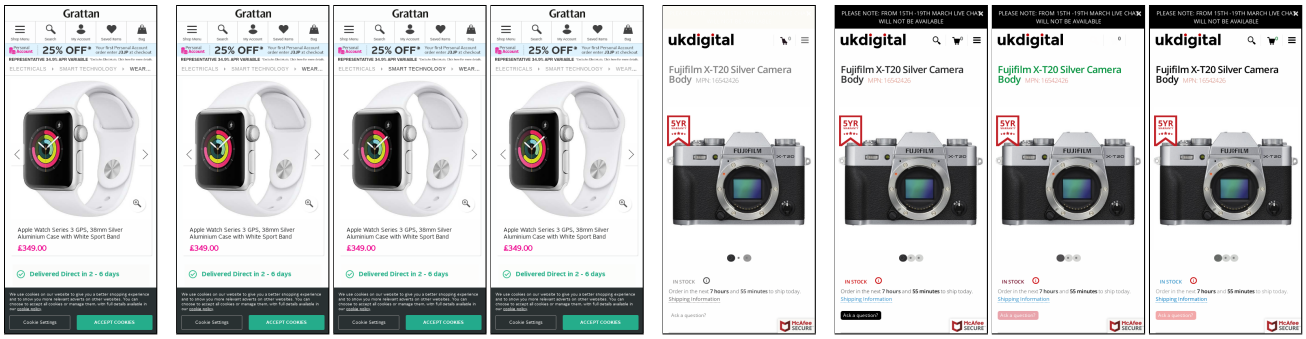
Ground-truth | Stats (mode) | Stats (sampling) | Ground-truth | Stats (mode) | Stats (sampling)



Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2] | Ground-truth (grayscale) | ColTran [2] (colorized) | ColTran [2]

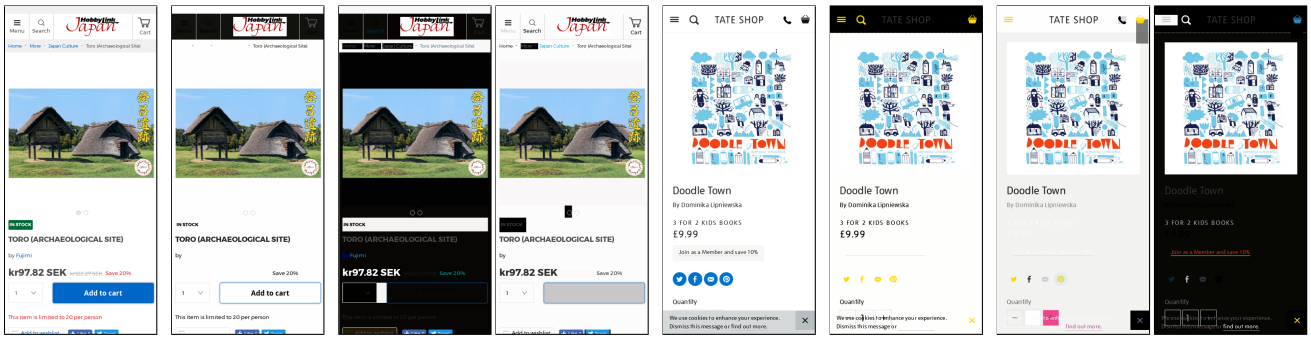


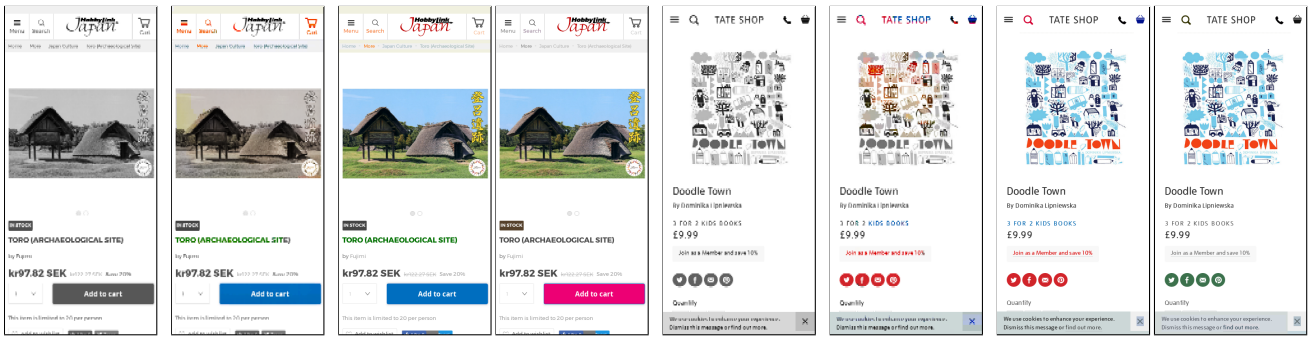
AR (greedy) | AR (top-p, p=0.9) | AR (greedy) | AR (top-p, p=0.9)

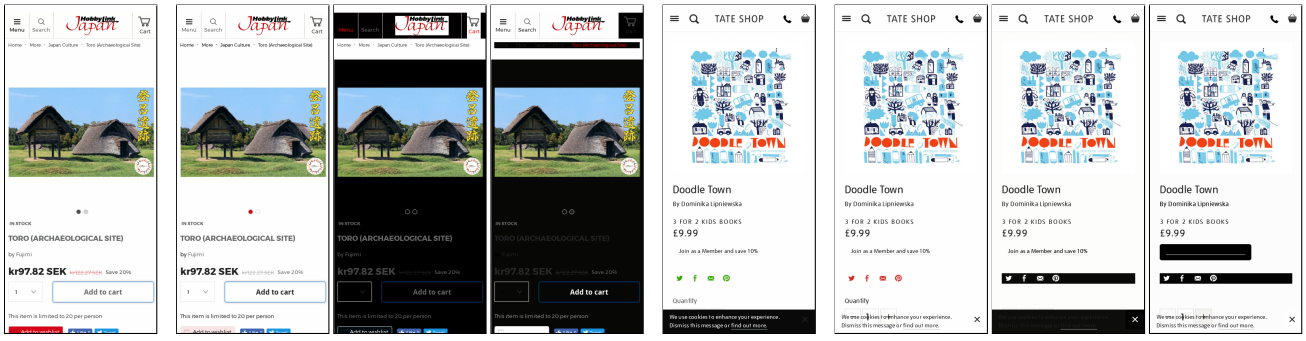


NAR | CVAE | NAR | CVAE

Figure 4: Additional qualitative results (4).







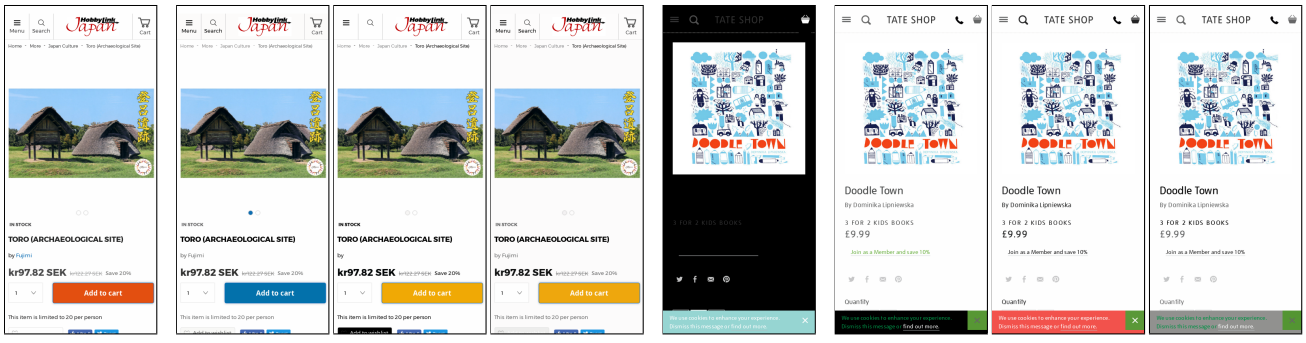


Figure 5: Additional qualitative results (5).